



UltraChat *C*Soft

Enhanced Chat System

UltraChat Entertainment Chat System

Version 1.0

User Manual

An Option Module for TBBS Version 2.2

**by Bob Hartman
and Philip L. Becker**

**Copyright © 1994 by Philip L. Becker, Ltd.
All Rights Reserved**

WARRANTY

UltraChat is manufactured by eSoft, Inc. We have made every effort to provide the user with error-free data diskettes. Should there be difficulty in transferring the data from these diskettes to your operating system diskette, contact us at (303) 699-6565 (voice) for replacement.

UltraChat is distributed on an "AS IS" basis only, without warranty. Neither eSoft, Inc. nor its authorized dealers shall have liability or responsibility to any person or entity with respect to liability, loss, or damage caused or alleged to be caused by this software. This includes, but is not limited to, any interruption of service, loss of business or anticipatory profits, or consequential damage resulting from the use of this software.

eSoft SOFTWARE SUPPORT

eSoft, Inc., provides a 24-hour-a-day multi-line support BBS service to registered TBBS system owners at (303) 699-8222 (eSoft Software Support BBS - speeds up to 28,800 bps supported). We strongly urge that this telecommunications system be used for error reporting and troubleshooting assistance. Because all system corrections will be distributed first via the eSoft Software Support BBS, we recommend that all registered licensees log on to the Support BBS at least once a month to receive latest system update information. (At your first logon, you will need to register the serial number of your TBBS master disk(s) and supply a password, which will be approved for full support access during the next working day. Other information is given online during the registration process.)

The UltraChat software is distributed without copy prevention protection, since we feel that such protection drastically reduces the utility of a program and punishes most the honest purchaser. We do, however, expect purchasers of UltraChat to honor their license agreement. This software is copyrighted, and licensed to the purchaser for your individual and exclusive use on a single CPU at a time. Any reproduction for use by other persons is a violation of our copyright and your license agreement.

THIS SOFTWARE IS NOT FOR SALE

eSoft, Inc. does not "sell" UltraChat. It sells only the media it is contained on. It licenses you the use of the software only under the following license terms and conditions.

License Agreement

Carefully read the following terms and conditions. Use of this product constitutes your acceptance of the terms and conditions, and your agreement to abide by them subject to paragraph 7 below.

1. **This is an end-user license.** You, the original purchaser, are granted this license for the use of the UltraChat software under the terms stated in this agreement. You may not assign or transfer the software or this license to any other person without the express written consent of eSoft, Inc. Any attempt to sublicense, assign, or transfer any of the rights, duties, or obligations hereunder is void. eSoft, Inc. does allow you to transfer this license under the conditions outlined on your registration form. This procedure constitutes express written consent under this provision if it is followed properly.
2. **The UltraChat software is copyrighted material.** Once you have paid the required single copy license fee, you may use the software as long as you like provided you do not violate the copyright or any of the following conditions.
3. **Single CPU License.** You may use the software on any computer for which it is designed so long as it is not in use on more than one computer at the same time. You must pay for additional licenses if you want to use this software on more than one computer at the same time.
4. **Backup Copies.** You may make as many backup copies of the software as you require to avoid loss. You are responsible for all backup copies you make, and must assure they do not result in any use of the software which would conflict with the provisions of paragraph 2 above.

5. **Software Modification.** You may not make any changes or modifications to the Licensed software not expressly authorized by eSoft, Inc., Philip L. Becker, Ltd. or their agents. This includes but is not limited to disassembly and reverse engineering the software. The single exception granted under this license is the changing of text strings in the programs for customized presentation.
6. **Federal Government.** This Software is Commercial Computer Software under the Federal Government Acquisition Regulations and agency supplements to them. The Software is provided to the Federal Government and its agencies only under the Restricted Rights Provisions of the Federal Acquisition Regulations applicable to commercial computer software developed at private expense and not in the public domain.
7. **You may refuse to abide by this license** by returning all materials within 30 days, along with a written statement that you have kept no copies of the software or documentation. This statement must be signed by you and becomes a legally binding statement that you have indeed destroyed any backup copies you may have made in those thirty days. If you keep the materials beyond the 30 day period, or refuse to assure that you have not kept any copies of the software or its documentation, then you are fully bound by this agreement.
8. **Limitation of Liability.** In no case shall the Liability of eSoft, Inc. or Philip L. Becker, Ltd. exceed the license fees paid for the right to use this software or One Hundred Dollars (\$100.00), whichever is greater.
9. **Dual Media.** This software is provided on dual media for your convenience. The software on both types of disk media is identical, and is covered by this single license. You are not authorized to distribute the "spare" disk to another party.
10. **This agreement may not be modified** except by a written instrument signed by eSoft, Inc. This license constitutes the entire agreement and understanding between you and eSoft, Inc., and supersedes any prior agreement or understanding whether oral or written relating to the subject of this License.

Table of Contents

An Introduction to UltraChat.....	1-1
UltraChat Initialization	1-7
Accessing UltraChat.....	1-8
Forcing the Use of Handles Within Chat.....	1-10
Separating Users Into Groups.....	1-10
Passing Parameters to UltraChat.....	1-11
"One-Shot" Access To UltraChat Functions.....	1-12
Suppress Keystroke Request on Chat Exit.....	1-13
Customizing the Who's Online Display.....	1-13
Why eSoft Created UltraChat	2-1
An UltraChat Orientation.....	2-3
General features of UltraChat	2-3
UltraChat Customization.....	2-8
How Commands and Elements Are Connected.....	2-10
Tutorial – Recreating TBBS Chat Using UltraChat.....	2-12
Creating the Configuration File.....	2-12
Entering Chat.....	2-14
Providing a Way Out.....	2-14
Exiting From Private to Public Chat	2-16
Listing Who's Online	2-17
Letting Users Communicate Privately.....	2-19
Summary.....	2-21
Optimizing UltraChat Performance.....	2-21
UltraChat Configuration Reference.....	3-1
The Parts of UltraChat Configuration	3-1
Command Formatting.....	3-1
Command Element Formatting	3-3
Command Element Sharing	3-4
General Configuration Statements.....	3-5
UCHAT_DIR <directory>	3-5
INCLUDE <filename>	3-6
NEW_USERS_NUM <number>	3-6
ANNOUNCE_CHARS_NUM <number>	3-6
MIN_CHANNEL_NUM <number>	3-6
MAX_CHANNEL_NUM <number>	3-7
PUB_CHANNEL_NUM <number>	3-7
DEFINE_CONF <number> <name>	3-7
LAST_LINE_NUM <number>	3-7
PUBLIC_ENTRY_KEY <key>	3-8

Table of Contents

PUBLIC_EXIT_KEY <key>	3-8
STOP_KEY <key>	3-8
PAUSE_KEY <key>	3-8
YES_KEY <key>	3-8
NO_KEY <key>	3-8
TBBS_PARSER	3-9
NAME_NUM_PARSER	3-9
SHORTCUT_KEY <char>	3-9
KILL_LINE_KEY <char>	3-10
OVERALL_TIMER <minutes>	3-10
INACTIVITY_TIMER <minutes>	3-10
ENTER_STRING <text>	3-11
HOT_KEYS <mode>	3-11
MATCH_ONLINE_FLAG <a_flags>	3-12
MATCH_USERLOG_FLAG <a_flags>	3-12
LIST_ME_SWITCH ON OFF	3-13
LIST_PRIVATE_SWITCH ON OFF	3-13
LIST_INVISIBLE_SWITCH ON OFF	3-13
READONLY_SWITCH ON OFF	3-13
Text Configuration Statements	3-14
SYSTEM_NAME_TEXT <text>	3-14
SYSTEM_ID_TEXT <id>	3-14
SYSOP_NAME_TEXT <name>	3-14
WAITING_TEXT <string>	3-14
IS_TEXT <string>	3-14
ARE_TEXT <string>	3-15
UNLIMITED_TEXT <string>	3-15
IGNORED_TEXT <string>	3-15
INVITED_TEXT <string>	3-15
User Related Text Configuration Statements	3-16
USER#01_TEXT <text>	3-16
USER#02_TEXT <text>	3-16
USER#03_TEXT <text>	3-16
USER#04_TEXT <text>	3-16
General Chat Text Configuration Statements	3-17
CHAT#01_TEXT <text>	3-17
CHAT#02_TEXT <text>	3-17
CHAT#03_TEXT <text>	3-17
CHAT#04_TEXT <text>	3-17
CHAT#05_TEXT <text>	3-18
CHAT#06_TEXT <text>	3-18
CHAT#07_TEXT <text>	3-18
CHAT#08_TEXT <text>	3-18
CHAT#09_TEXT <text>	3-19

CHAT#10_TEXT <text>	3-19
CHAT#11_TEXT <text>	3-19
CHAT#12_TEXT <text>	3-19
CHAT#13_TEXT <text>	3-19
CHAT#14_TEXT <text>	3-20
CHAT#15_TEXT <text>	3-20
CHAT#16_TEXT <text>	3-20
CHAT#17_TEXT <text>	3-20
CHAT#18_TEXT <text>	3-20
CHAT#19_TEXT <text>	3-20
CHAT#20_TEXT <text>	3-21
CHAT#21_TEXT <text>	3-21
CHAT#22_TEXT <text>	3-21
CHAT#23_TEXT <text>	3-21
CHAT#24_TEXT <text>	3-21
General Error Text Configuration Statements	3-22
ERROR#01_TEXT <text>	3-22
ERROR#02_TEXT <text>	3-22
ERROR#03_TEXT <text>	3-22
ERROR#04_TEXT <text>	3-22
ERROR#05_TEXT <text>	3-22
ERROR#06_TEXT <text>	3-23
ERROR#07_TEXT <text>	3-23
ERROR#08_TEXT <text>	3-23
Miscellaneous Text Configuration Statements	3-23
MISC#01_TEXT <text>	3-23
MISC#02_TEXT <text>	3-24
MISC#03_TEXT <text>	3-24
MISC#04_TEXT <text>	3-24
MISC#05_TEXT <text>	3-24
MISC#06_TEXT <text>	3-24
MISC#07_TEXT <text>	3-24
MISC#08_TEXT <text>	3-25
MISC#09_TEXT <text>	3-25
MISC#10_TEXT <text>	3-25
MISC#11_TEXT <text>	3-25
MISC#12_TEXT <text>	3-25
MISC#13_TEXT <text>	3-25
MISC#14_TEXT <text>	3-25
MISC#15_TEXT <text>	3-25
MISC#16_TEXT <text>	3-26
Color Changing Commands	3-26
COLOR#01_TEXT <ansi_sequence>	3-26
COLOR#02_TEXT <ansi_sequence>	3-26

Table of Contents

COLOR#03_TEXT <ansi_sequence>	3-26
COLOR#04_TEXT <ansi_sequence>	3-26
Channel Changing Commands.....	3-27
CHANNEL#01_CMD	3-27
CHANNEL#02_CMD	3-27
CHANNEL#03_CMD	3-28
CHANNEL#04_CMD	3-28
CHANNEL#05_CMD	3-28
CHANNEL#06_CMD	3-28
CHANNEL#07_CMD	3-28
CHANNEL#xx_CONFIRM	3-29
CHANNEL#xx_CHANPMT	3-29
CHANNEL#xx_PWPMT.....	3-29
CHANNEL#xx_ENTRY.....	3-29
CHANNEL#xx_EXIT.....	3-29
CHANNEL#xx_BADNUM.....	3-29
CHANNEL#xx_NOTNUM.....	3-30
CHANNEL#xx_NOTINV.....	3-30
CHANNEL#xx_BADPW.....	3-30
General Chat Commands.....	3-30
CHAT#01_CMD.....	3-31
CHAT#02_CMD.....	3-31
CHAT#03_CMD.....	3-31
CHAT#04_CMD.....	3-31
CHAT#05_CMD.....	3-32
CHAT#06_CMD.....	3-32
CHAT#xx_CONFIRM.....	3-32
CHAT#xx_PROMPT	3-32
CHAT#xx_EXIT.....	3-32
CHAT#xx_REMOTE	3-33
CHAT#xx_NOCHAT.....	3-33
CHAT#xx_SELF.....	3-33
Private Message Commands	3-33
PRIVATE#01_CMD - PRIVATE#09_CMD.....	3-34
PRIVATE#10_CMD.....	3-34
PRIVATE#xx_CONFIRM.....	3-34
PRIVATE#xx_NAMEPMT	3-35
PRIVATE#xx_MSGPMT.....	3-35
PRIVATE#xx_EMPTY	3-35
PRIVATE#xx_REMOTE	3-35
PRIVATE#xx_NOCHAT	3-35
PRIVATE#xx_ERROR	3-36
Who's Online Commands.....	3-36
WHO#01_CMD - WHO#06_CMD.....	3-37

WHO#07_CMD	3-37
WHO#08_CMD	3-37
WHO#09_CMD	3-37
WHO#10_CMD	3-37
WHO#11_CMD	3-38
WHO#12_CMD	3-38
WHO#13_CMD	3-38
WHO#xx_HEADER	3-38
WHO#xx_TOTAL	3-38
WHO#xx_NORMAL	3-38
WHO#xx_IDLE	3-39
WHO#xx_INVISIBLE	3-39
WHO#xx_PAUSE	3-39
WHO#xx_PUBLIC	3-39
WHO#xx_MYCHAN	3-40
WHO#xx_HISCHAN	3-40
WHO#xx_PRIVATE	3-40
WHO#xx_NOUSERS	3-40
WHO#xx_NAMEPMT	3-41
WHO#xx_ERROR	3-41
WHO#xx_ANON	3-41
Chat Exit Commands	3-41
EXIT#01_CMD	3-41
EXIT#02_CMD	3-42
EXIT#03_CMD	3-42
EXIT#04_CMD	3-42
Help File Display Commands	3-42
HELP#01_CMD - HELP#09_CMD	3-43
HELP#xx_FILE <filename>	3-43
HELP#xx_ERROR	3-43
Text Display Commands	3-43
TYPE#01_CMD - TYPE#19_CMD	3-44
TYPE#xx_TEXT	3-44
Status Toggle (On to Off, Off to On) Commands	3-44
TOGGLE#01_CMD	3-45
TOGGLE#02_CMD	3-45
TOGGLE#03_CMD	3-45
TOGGLE#04_CMD	3-45
TOGGLE#05_CMD	3-45
TOGGLE#xx_ON	3-46
TOGGLE#xx_OFF	3-46
Status Forced On (Enable) Commands	3-46
ON#01_CMD	3-46
ON#02_CMD	3-47

Table of Contents

ON#03_CMD	3-47
ON#04_CMD	3-47
ON#05_CMD	3-47
ON#xx_TEXT	3-48
Status Forced Off (Disable) Commands.....	3-48
OFF#01_CMD.....	3-48
OFF#02_CMD.....	3-48
OFF#03_CMD.....	3-49
OFF#04_CMD.....	3-49
OFF#05_CMD.....	3-49
OFF#xx_TEXT	3-49
Three-Line Bio Commands.....	3-50
BIO#01_CMD.....	3-50
BIO#02_CMD.....	3-50
BIO#xx_YN.....	3-50
BIO#xx_PROMPT	3-51
Edit Commands.....	3-51
EDIT#01_CMD.....	3-51
EDIT#02_CMD.....	3-51
EDIT#03_CMD.....	3-52
EDIT#04_CMD.....	3-52
EDIT#xx_PROMPT	3-52
EDIT#xx_CONFIRM.....	3-52
EDIT#xx_ERROR.....	3-52
Interaction Commands.....	3-53
INTERACT#01_CMD	3-53
INTERACT#02_CMD	3-54
INTERACT#03_CMD	3-54
INTERACT#04_CMD	3-54
INTERACT#05_CMD	3-54
INTERACT#06_CMD	3-55
INTERACT#07_CMD	3-55
INTERACT#xx_ON.....	3-55
INTERACT#xx_ON2.....	3-55
INTERACT#xx_OFF	3-56
INTERACT#xx_OFF2	3-56
INTERACT#xx_ALLON.....	3-56
INTERACT#xx_ALLOFF	3-56
INTERACT#xx_NAMEPMT.....	3-57
INTERACT#xx_REMOVE	3-57
INTERACT#xx_REMOTEINUC	3-57
INTERACT#xx_REMOTE	3-57
INTERACT#xx_ERROR	3-58
Action Commands.....	3-58

ACTION#01_CMD – ACTION#09_CMD.....	3-59
ACTION#xx_CONFIRM.....	3-59
ACTION#xx_REMOTE.....	3-59
Executing TBBS Menu Commands.....	3-59
DOTBBS#01_CMD – DOTBBS#09_CMD.....	3-60
DOTBBS#xx_TYPE.....	3-60
DOTBBS#xx_DATA.....	3-60
Moan Macros.....	3-61
DEFAULT_RESPONSE <string>.....	3-62
DEFAULT_SENDS <string>.....	3-62
IF_USER_RESPONSE <string>.....	3-62
IF_USER_SENDS_USER <string>.....	3-62
IF_USER_SENDS <string>.....	3-62
IF_MSG_RESPONSE <string>.....	3-62
IF_MSG_SENDS <string>.....	3-63
IF_USER&MSG_RESPONSE <string>.....	3-63
IF_USER&MSG_SENDS_USER <string>.....	3-63
IF_USER&MSG_SENDS <string>.....	3-63
Using Text in UltraChat Command Elements.....	3-64
Wrapping Text Line-to-Line.....	3-65
Adding Carriage Returns and Control Characters.....	3-65
Adding ANSI Escape Sequences.....	3-66
Restricting Configurations Within UltraChat.....	3-67
P# = <param>.....	3-68
GROUP = <group>.....	3-68
NAME = <name>.....	3-68
LINE = <lines>.....	3-68
LANG = <c>.....	3-69
PRIV <conditional> <priv_level>.....	3-69
BAUD <conditional> <baud_rate>.....	3-69
DAYS = <day>.....	3-70
DATE = <date>.....	3-70
Nesting Restrict Blocks.....	3-71
Insertion Parameters.....	3-72
Insertion Parameter Formatting.....	3-73
General Insertion Parameters.....	3-73
Remote (Target) User Insertion Parameters.....	3-76
Userlog Image Insertion Parameters.....	3-77
An Introduction to UltraChat Linking.....	4-1
The UltraChat Link Script Language.....	4-1
Glossary.....	A-1

An Introduction to UltraChat

What you hold in your hands is the most powerful entertainment chat software ever developed for use on bulletin board systems. UltraChat is the product of over a year's worth of design and development, representing tens of thousands of man hours of effort. UltraChat brings the configurability for which TBBS is known to the chat arena, offering unparalleled flexibility and control. Like TBBS itself, UltraChat is a complex product that requires time, effort and patience to master fully. The benefit to you and your users is clear – UltraChat can be made to look and function like the chat facilities of virtually any BBS. You can even design your own totally unique chat system, with the functions, keystrokes, prompts and responses you choose.

UltraChat is shipped to you with several sample configurations that should be suitable for virtually any operating BBS. By simply installing UltraChat and one or more of the sample configurations, you can begin offering the benefits of UltraChat with a minimal amount of effort. Then, as your schedule and level of inclination allow, you can begin tapping UltraChat's more advanced features and configurability options.

Thank you for selecting UltraChat. We hope you and your users enjoy the product as much as we enjoyed bringing it to you.

IMPORTANT: Make sure you view the README.DOC file, located in the root directory of the release disk, for late-breaking information, corrections, and other important details which are not reflected in this manual.

Installation of UltraChat

UltraChat installation consists of the following steps:

1. Physically copy the software onto your hard disk.
2. Reference UltraChat on the TBBS command line.
3. Install the supplied patch to TBBS using UPDATE (if you're running TBBS Version 2.2). Please note that UltraChat will not run on versions of TBBS earlier than 2.2, nor will it run on single line versions of TBBS.
4. Create a directory for temporary UltraChat files
5. Using the supplied UltraChat configuration file (or after you create your own) compile the configuration using UltraChat
6. Set up your TBBS menus to access UltraChat

Now let's look at each of these steps in detail...

To install the UltraChat option module, copy all files from the root directory of your UltraChat release disk into the TBBS sub-directory on your hard disk.

Next add the following command line switch to your MLTBBS command line:

```
/O:UCHAT
```

If you already have option modules installed on your TBBS (such as TDBS or SYSOM) then simply add the reference to UltraChat at the end of the list, like so:

```
/O:TDBSOM, SYSOM, UCHAT
```

This will install the UltraChat option module. Note: If you receive the error message "Not enough memory" when you run TBBS you should change to using /OX to load option modules into EMS memory. In the above example this would be:

```
/OX:TDBSOM, SYSOM, UCHAT
```


Next, **ONLY IF YOU ARE RUNNING TBBS VERSION 2.2**, use the UPDATE program (available from the cSoft Support Board if you do not already have a copy) to apply a special patch to TBBS that allows UltraChat to better integrate into your TBBS system.

From DOS, in your TBBS directory (where UPDATE.EXE should be placed), type the following:

```
UPDATE ALL /U:UCHAT.BIN
```

UPDATE will run, report some status messages, and indicate that it has completed updating your TBBS. **NOTE:** Versions of TBBS later than 2.2 do not require this update.

Next, create a directory that will be used for UltraChat's temporary data files. If you have a large RAM disk, UltraChat performs best if this directory is placed on a RAM disk drive. This directory should be capable of holding at least 16k per line that your system runs, so a 64 line system would require a directory that can hold approximately 1M of data. This directory name will be used in your UCHAT.CTL file as the UCHAT_DIR directory (a configuration statement discussed later on).

Next, compile the supplied UltraChat configuration file. Compiling means that the configuration for UltraChat, created and manipulated as a text file, is copied in a special format to a new file that UltraChat can use online; the configuration file must be compiled before UltraChat can run. You will probably wish to at least look at this file before compiling it. **YOU WILL NEED TO CUSTOMIZE THIS FILE TO SOME DEGREE BEFORE MAKING ULTRACHAT AVAILABLE TO YOUR USERS!** If you wish to simply test and "play" with UltraChat, compile the supplied sample and come back to configuration later. Configuring UltraChat is covered elsewhere in this manual.

To compile, from DOS, in your TBBS directory type the following:

```
UCHAT COMPILE
```

Various status messages will be displayed, and it should indicate that compilation was completed without error. (If you have made changes, and the file no longer compiles, check your changes to make sure they are formatted properly; refer to the appropriate sections of this manual for help.)

HINT: If you insert the UCHAT COMPILE command in your RUNBBS.BAT batch file, just prior to the MLTBBS command you will insure that the UltraChat configuration is compiled and current each time TBBS runs.

Finally, set up menus in TBBS for UltraChat as described beginning on page 1-6.

Option Modules and Memory Usage

When you install an option module such as UltraChat, the total memory required by your TBBS system will increase. This increase occurs in two areas:

- Space to put the UltraChat software itself
- Space to put data manipulated by UltraChat

The UltraChat software itself normally resides in base (conventional) memory. This is the initial 640k of memory installed in a PC compatible. Therefore, more conventional memory is needed to store UltraChat. This is called OM CODE (for option module code) and is the UltraChat software application itself.

The data manipulated by UltraChat is called OM UDATA (for option module user data). There is one section of OM UDATA (called a segment) for every line defined in TBBS – whether the line is used by a caller or not. This memory can be located in base memory, in EMS (expanded) memory, or in a combination of both.

UltraChat requires the following additional memory:

OM CODE memory:	60,000 bytes (plus 600 bytes per line)
OM UDATA memory:	48,000 bytes per user conventional or 48 kbytes per user EMS

OM UDATA memory is shared with other option modules, such as TDBS, SYSOM, etc. The size of the OM UDATA will be no larger than the largest size requested by any one option module you have installed. The following illustration shows this concept:



The largest amount of OM UDATA memory required by any one option module is the maximum amount needed by all combined

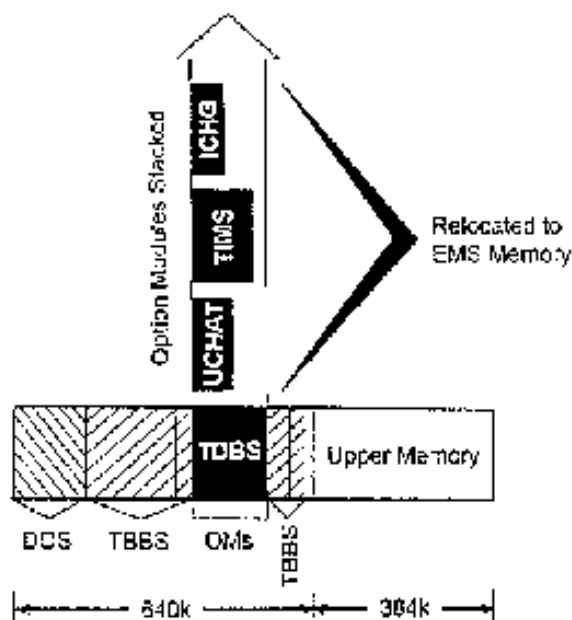
When There's Not Enough Memory Available

You may be unable to load your TBBS system after installing UltraChat due to inadequate available memory. Assuming you have EMS (expanded) memory installed in your machine (commonly done with utilities such as QEMM, 386Max, etc.) then you are most likely running out of base (conventional) memory. To help work through this situation, TBBS offers a technique called option module code stacking. On 386, 486 and Pentium based PCs that implement true LIM 4.0 EMS memory, TBBS can "stack" option modules on top of each other into EMS memory.

To enable option module code stacking in TBBS, simply change the /O: MLTBBS command line switch to /OX: like so:

```
/OX: TINS, TDBSOM, SYSOM, QSO, ICHG
```

With option module code stacking, the amount of conventional memory used for option module code (OM CODE) is no larger than the largest single option module you have installed (usually TBBS). No matter how many option modules you install, no additional conventional memory will be used – the OM CODE will be loaded into EMS memory instead. The following illustration shows this concept:



Of course, you must ensure that you have enough EMS memory installed and available before option module code stacking can work. If TBBS still fails to load after enabling code stacking, then check to make sure you have enough memory installed in your PC, and that your memory manager is configured properly to use the memory as EMS.

Adding UltraChat to TBBS Menus

When the UltraChat option module is installed, a new command type becomes available for use in TBBS menus. This command is like any other TBBS menu command and may be used in any menu as you wish. The new command is:

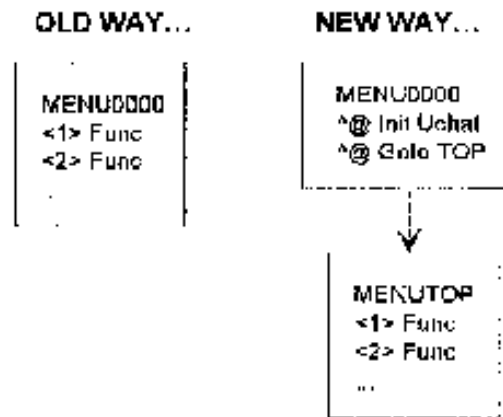
Type = 203

There are several Opt Data switches available that control the behavior of UltraChat. They are described elsewhere in this manual.

As with all TBBS commands, the Type 203 command may be used in as many different menu entries and in as many different forms as you wish. All normal TBBS security occurs through the privilege levels and access flags as it does with any TBBS menu entry.

UltraChat Initialization

UltraChat is unique in that it must be "initialized" each time a user connects to the system. To do this, you must execute UltraChat from an auto-executing menu entry in your top level menu, 0000, and then pass control to another menu. This illustration shows the concept of how this is done:



You may already have a structure like this in place. If you do not, then to change to the new structure, follow these steps:

- Rename your 0000 menu in TBBS to TOP by using the DOS command:

REN MENU0000 .CTL MENUTOP .CTL
- Create a new 0000 menu with MEDIT or SDL
- In this 0000 menu first place an auto-executing menu entry to initialize UltraChat
- In this 0000 menu next place an auto-executing menu entry of Type = 35 and Opt Data = TOP to transfer to your original TOP level menu and set it for the duration of the call as the top level menu.

NOTE: If you operate in multilingual mode, or have the RIP Kit installed, you will need to duplicate this hi-level menuing structure for each of the languages you have installed; menu 0000 for language A, menu 00LB for language B, menu 00LC for language C, etc. To initialize UltraChat, the auto-executing menu entry in the newly created 0000 should have these characteristics:

```

Type=203                ; Execute UltraChat
Opt Data=/I             ; /I means "initialize"
  
```

Here is what such a menu would look like in SDL:

```
Menu: 0000
Entry:
Key=^@ Type=203 Opt Data=/I

Entry:
Key=^@ Type=35 Opt Data=TOP
EndMenu:
```

This menu will initialize UltraChat, and force the user to enter their three-line "bio" information if they have not entered it previously.

You can initialize UltraChat as shown, or if you prefer, you can use one of several optional forms of initialization by using these Opt Data switches:

/I:1

This is the same as /I, except that a user is not required to fill out a three-line bio, useful for systems that do not allow chat access until after a user is a member.

/I:2

This is the same as /I, except that a "logon announcement" is not generated, useful for systems not wishing to have logon announcements.

/I:3

This initializes the user without checking for a three-line bio, and without generating a logon announcement.

Accessing UltraChat

Once UltraChat is installed and set to initialize each time a user logs into the system, you need to provide TBBS menu entries for people to use the chat features.

To allow a user to enter UltraChat, create a menu entry where desired with the following characteristics:


```
Type=203                ; Execute UltraChat  
Opt Data=/E:2           ; /E means "enter"
```

Here is what such a menu entry might look like in SDL:

```
Entry: <C>hat Access  
Key=C Type=203 Opt Data=/E:2
```

You can allow entry to UltraChat as shown, or if you prefer, you can use one of several optional forms of entry by using these Opt Data switches:

/E:1

Allows entry "Oracomm style" where all lines in use are listed, and a user receives a question about whether they want private, public or sysop chat.

/E:2

Allows entry directly to the lowest channel number allowed with no question asked whatsoever.

/E:3

Allows entry "DLX style" where the user enters directly to public chat, but is prompted as to which channel they wish to enter.

/E:4

Allows entry directly to the last channel the user was in, if they can still go there. If they cannot go there, they are placed in the lowest numbered channel configured. Note this type of entry cannot return a user to private one-on-one style chat; only a chat channel (whether public or private).

/E:5,#

Allows direct entry to a specific channel number. The # parameter is the public chat channel number to enter. This can be used as a way to force different people into different chat sections, and if you do not provide a channel change command, to prohibit them from moving about.

`/E:6,#`

Allows direct entry to a named conference. The # parameter is the named conference identifier, and must match a defined named conference. Refer to the `DEFINE_CONF` configuration statement (discussed later on) for details on defining a named conference.

Forcing the Use of Handles Within Chat

You can force users to use a handle – a name different from their user name or ID – within the chat environment. (Refer to the glossary in Appendix A for more information on handles.)

Whenever you want to force the use of handles, simply add the `/H` switch to UltraChat's Opt Data line.

IMPORTANT: Although not a requirement, the `/H` switch is best used when UltraChat is initialized (see the `/I` switch, above). This will force a user to enter a handle before the logon announcement is issued.

Here is what an initialization menu entry might look like in SDL when handles are forced:

Entry:

Key=^@ Type=203 Opt Data=/I /H

Separating Users Into Groups

In many cases, you may wish to logically separate users into unique groups for the purposes of UltraChat related functions (chat, who's online displays, three-line bio lookup, and so on). For example, if you operate a single BBS with a general public section and a user club section, you might not want users of one part of the system to interact with users of the other part(s). UltraChat provides group capabilities to allow this separation. Refer to the glossary in Appendix A for more details about groups.

To place a user within a group, use this switch:

`/G:<id>`

This Opt Data switch marks a user as being a member of a particular group, where `<id>` is the group identifier. The group identifier may be any string of four (4) characters or less. When a user is assigned to a group, they cannot perform any UltraChat functions directed at users outside of their group. The only exception is the special group `<id>` of “0” that essentially removes the user from a group. This is also the default group; should you not use the `/G:` switch, all users are in the same group.

This switch is particularly useful in conjunction with the `/I` initialization switch to mark a user as a member of a group from the moment they login to the system. (When used this way, the `/G: <id>` switch must be used before the `/I` in the Opt Data, as shown below.)

SDL menu example (at initialization):

Entry:

```
Key=~@ Type=203 Opt Data=/G:BASE /I
```

Note that you may configure up to 250 groups, plus the default “all included” group.

Passing Parameters to UltraChat

UltraChat allows you to pass parameters – text information – from the menu entry to UltraChat upon entry to the chat environment. These parameters can then be used to dynamically change UltraChat behavior based on the parameter(s) passed.

To pass a parameter, use the Opt Data switch:

```
/P<#>:<string>
```

Where `<#>` is a number between 1 and 9, and `<string>` is any text string enclosed in single or double quotes. The parameters you pass in this fashion can then be used by the UltraChat configuration file to change the behavior of UltraChat for a particular user or group of users.

Here's an example of parameter passing in an SDL menu entry:

Entry:

<M>ajor BBS Chat Emulation

Key=M Type=203 Opt Data=/P1:"MBBS" /E:2

This would pass the string "MBBS" as parameter 1 to UltraChat. Within the UltraChat configuration file, you can use the P# restrict type in conjunction with RESTRICT blocks to control configuration based on a passed parameter. For example this parameter causes UltraChat to emulate a Major BBS system's chat facilities with the sample configuration file supplied. Refer to the section "Restricting Configurations Within UltraChat" later in the manual for details.

"One-Shot" Access To UltraChat Functions

UltraChat allows you to create menu entries that directly access a single chat command and then return to the TBBS menu. This is called a "one-shot" since the function is executed singly and then UltraChat exits. To create such a menu entry use Opt Data switch:

/C:<string>

The <string> is sent to UltraChat and is executed exactly as though the user had entered chat and then typed in the string. Therefore, the <string> needs to consist of a command that is valid and legal for that user. To ensure that UltraChat correctly extracts the command place the string in single or double quotes.

SDL menu examples:

Entry:

<S>end a Message to Another User

Key=S Type=203 Opt Data=/C: "/T"

Entry:

<P>age the Sysop

Key=P Type=203

Opt Data=/C: "/T Sysop Please come chat!"

NOTE: In the above example, the /T command is the "Oracomn style" command to send a message to another user; these examples assume a configuration that is "Oracomn like."

Suppress Keystroke Request on Chat Exit

When you execute an UltraChat "one-shot" (refer to the /C: switch, discussed previously), UltraChat will display any appropriate command elements and then pause with a "press any key to continue" prompt before exiting back to TBBS.

For some commands, such as a handle change, this is the desired effect. Some one-shot commands, however, are best executed with an immediate return to TBBS without the pause. When you want this behavior use the /N switch in conjunction with the /C: switch as follows:

SDL menu example:

```
Entry: <S>end a Message to Another User  
Key=S Type=203 Opt Data=/C: "/T" /N
```

Customizing the Who's Online Display

One of UltraChat's primary features is the who's online display. Several such displays are available. In most types of who's online displays, UltraChat will display a text string that identifies what a user is doing at any given moment. By default UltraChat assigns a fixed text description for each possible TBBS command type. However, you can dynamically override and customize the text shown to fully customize your "who's online" display by using a special UltraChat Opt Data switch as follows:

```
/D: < # > , <text1> , <text2>
```

where the < # > portion of the switch designates a TBBS command (the Type = field in a TBBS menu entry) to execute.

The <text1> portion of the switch is the Opt Data string for this command. It must be in quotes if there are any embedded spaces.

The <text2> portion of the switch provides the text to show in replacement of the %RCOMMAND% insertion parameter in who's online displays as long as the user remains within the command type provided by the < # > portion of the switch. It must be contained in quotes if there are embedded spaces. This text is limited to a maximum of 30 characters.

By executing UltraChat with this switch instead of executing a TBBS command directly, you can tailor to as high a degree as you want what is shown in the UltraChat "who's online" displays.

For example, if a menu entry looked like this:

```
Entry:
<E>nter The Wild Side:
Type=203
OptData=/D:200,"F:\TBBS\TWS /Q","Wild Side!"
```

then UltraChat (represented by Type = 203) would be executed. It, in turn, would set the <text2> string "Wild Side!" as the %RCOMMAND% text and then execute a TBBS Type 200 (the number given by the <#> portion of the switch), with an Opt Data of "F:\TBBS\TWS /Q" (the <text1> portion of the switch).

Here's another example:

```
Entry:
<S>YSOM
Type=203 Opt Data=/D:205,"", "Using SYSOM"
```

This would cause UltraChat to set "Using SYSOM" as the %RCOMMAND% text, then execute a Type 205 (SYSOM, in this case) with blank Opt Data (since the <text1> string is empty).

Note that UltraChat understands what a user is doing based only on the TBBS menu command type the user is executing. Once you set a string in conjunction with a particular command type number, that string will be displayed for that user every time they execute that command type until you override it again. For example, in the first case outlined above, "Wild Side!" would be displayed for a Type 200 for that user until you override the text again using the same technique.

IMPORTANT: If you want either the <text1> or the <text2> portion to be blank, you must use a pair of double quotes, like this - "" - to designate an empty text string.

Sample Emulations

UltraChat is supplied with four sample emulations: one each for Oracomm, Major BBS, DIX and TBBS. These closely emulate the chat facilities of these various BBS packages, and are designed to make the transition to UltraChat (and/or TBBS) smoother for you and your users. They can continue to use and enjoy the same functions, invoked with the same keystrokes they are familiar with, while within the UltraChat environment.

IMPORTANT NOTE: These sample emulations are for UltraChat. Although they will help serve to make your overall BBS more familiar when converting from another BBS package to TBBS/UltraChat, their primary benefits will be available only within the chat environment. Outside of the chat environment, you will still need to design your system (using TBBS menu designs and/or changes to the TBBS language file) as you or your users desire, and if you wish, to pattern the system functions and keystrokes after the BBS software you are converting from.

These emulations are provided as separate configuration files for UltraChat. You can use any one or more of the emulations. Many TBBS sysops have decided to make all of them available to users simultaneously and allow users to pick which one they like best. Or, if you wish, you could settle for one of the emulations and make additions and modifications to it as you see fit. With UltraChat, as with TBBS, you are under complete control of the system and the services it provides.

When you are converting from another BBS product to TBBS/UltraChat, in most cases, you'll select the emulation that matches the product you're converting from. Initially, you may wish to simply run that emulation as-is. eSoft has put a considerable amount of time into making these emulations match the chat functions of the products emulated as much as technically possible. Or you may wish to make changes to the emulation, or add functions that the product you converted from cannot offer. During beta testing of the UltraChat product, most sysops found their users enjoyed having additional, previously unavailable chat functions open to them. As you explore UltraChat, make notes of its capabilities that you may wish to add to your system.

The emulations are packaged on the UltraChat disk in individual sub-directories. Within each sub-directory is a configuration file for the emulation, as well as a collection of help files that go with the emulation.

CAUTION: In cases where you wish to use multiple emulations simultaneously, you will need to ensure that the names of the various files for each emulation do not conflict with the files of another emulation.

The directories and their emulations are:

- \DLX Emulates DLX chat facilities
- \ORACOMM Emulates Oracomm chat facilities
- \MAJOR Emulates Major BBS chat facilities
- \TBBS Emulates TBBS chat facilities

Converting From Another BBS Package

Many sysops are converting to TBBS/UltraChat from other BBS packages. They have found, as you might have, that TBBS allows them to offer chat facilities compatible with the BBS they have converted from, while giving them a platform which can support many additional functions (such as fully integrated FidoNet support via TIMS, fully integrated Internet mail support via PIMP, online games and system utilities via TDBS, QWK offline reader support via QSO, and so on).

However, converting from one BBS package to another can be a traumatic experience for users of your BBS. By definition, no BBS package is capable of emulating 100% another BBS package's interface, keystrokes, capabilities, or functions. TBBS is the most configurable BBS product available anywhere, and this flexibility makes it a superior platform when converting from another package. You must understand, however, that even a BBS as configurable and powerful as TBBS cannot be made to match your old system exactly.

During the development and testing of UltraChat, eSoft worked with dozens of entrepreneurial sysops who converted from other BBS packages to TBBS/UltraChat. From this experience, we learned that in most cases sysops did in fact experience initial shrinkage in user base, new user subscriptions, and income. But in all cases, the trend reversed itself after being on TBBS and UltraChat for just a short time. Virtually all the sysops who converted reported a steady (and in some cases very dramatic) increase in user base, new subscriptions, and most importantly, income after the conversion was complete.

This did not go unnoticed by other sysops. In fact, we received a number of comments from sysops who were not beta testing UltraChat indicating that their business was being impacted by another system who was beta testing the product!

This process can be alarming at first, but it is an understandable dynamic. Most users enjoy your BBS for very intangible reasons, often being quite emotionally bonded to your system. When you make a major change to "their baby," initial fear and disappointment is inevitable. You may receive complaints – in some cases lots of them – but do not allow yourself to get discouraged. Provided you execute a conversion plan carefully (which we'll discuss below), you'll be handsomely rewarded in time.

If you are converting from another BBS package, keep the following in mind:

- **Develop a conversion plan.** This includes a timetable for the conversion from the old software, as well as a sort of "marketing plan" – how you'll prepare your users for the change.
- **Allow enough time.** When developing your plan, allow yourself plenty of time to learn TBBS and UltraChat before you put it online.
- **Do not convert before you understand.** Make absolutely certain that you are proficient with TBBS before you convert. If problems occur during the conversion you will then be equipped with enough knowledge to solve them. If you try to rush the conversion, you very well might end up in a tight situation you cannot easily recover from, which will serve to annoy your users and impact your income as well.

- **Build first, then convert.** On a separate system, install and develop the new TBBS software. Get everything installed and working perfectly. Test the new system with various user classes. Test as a new user. Make absolutely certain that everything on the system from top to bottom is working properly. Many sysops who participated in the testing of UltraChat went so far as to have extra lines installed and put their new TBBS online on a separate telephone number group before conversion. This allowed users to get familiar with the new system before the old one went away, and provided the sysop with a real service by having the new system tested in a "real world" setting.
- **Handle users carefully.** Before you make the change, notify your users in advance through online bulletins, mailed information, or whatever you feel is best. Let them know you're making a change, and most importantly, tell them why. Be positive! Tell them all the good things they're going to get from making this change. Outline everything new you'll be adding: online games, networked mail (FidoNet, Internet), QWK offline reader support, and so on – anything that's a new feature you can now provide and could not offer before. If users are excited about the change, they'll accept it much more readily.
- **Be prepared to help.** Once the conversion is complete, be prepared to spend some long sysop hours helping people with the new system. There are bound to be questions, and depending on how much time you invested up front, there may be a problem or two as well. Be patient with yourself and your users. Things will calm down and return to normal in time.
- **Perform your conversion during the week (not the weekend),** and during the times of day that eSoft technical support staff is available. (At this writing, our hours are Monday through Friday, from 10:00am to 4:30pm Mountain Time, except major US holidays.) That way, if you do need the assistance of eSoft technical staff, you can reach us as the conversion is implemented.

Why eSoft Created UltraChat

UltraChat is one of the most ambitious product development projects that eSoft has ever undertaken. It represents thousands of man-hours of effort in research, design, development, and testing, over the course of nearly a year and a half. We are often asked why we decided to create UltraChat and devote so many resources to the project.

Historically, TBBS has been used in business and entrepreneurial settings where chat facilities are either not an issue at all, or have been just a small part of the overall focus of the system. In 1992, we began to notice a marked increase in the number of potential customers who were approaching us needing a BBS package with superior chat facilities. Simultaneously, our current customers were asking for more and more in the way of a chat system for TBBS. We decided to embark on a research project to determine why there was such a change in the tide of demand for chat facilities.

What we discovered were two primary reasons: lack of availability, and popularity.

Sysops who were running Oracom and DLX bulletin boards were finding that they had too much success. These packages have a "hard limit" of 32 simultaneous phone lines, with no availability for expansion – but with increasing demand for the system. We talked to entrepreneurial sysops who conveyed horror stories of endless busy signals and user complaints about them. These sysops were stuck with too much success and no way to make the situation any better.

Meanwhile, chat as a BBS function was increasing in popularity as more and more members of the general public became aware of bulletin boards and their capabilities. Our own customers, who previously had been satisfied with TBBS' internal chat facilities, began to ask more and more often "when is eSoft going to improve the chat facilities of TBBS".

To address these needs, eSoft studied what it would take for us to create and offer improved chat facilities. Initially, we explored enhancing TBBS' own chat facilities. But our research had shown that truly good quality chat is not a small thing – it has many unique requirements. We also knew that the best chat would have to be

totally configurable, following the TBBS model. As we outlined what we would need to do, it became clear that adding capabilities to TBBS itself would be a mistake, unnecessarily bloating and complicating the core product, affecting everyone who used it.

We also found that while sysops were happy with their success, most felt their chat software could be improved. They were receptive to the idea of change if the software would do what they wanted.

We thus decided to begin development of a chat option module designed specifically for those sysops who needed enhanced chat facilities. We used the results of our research to develop a feature list for what would become UltraChat. Since our research had told us that chat is a very emotional, intangible thing for a BBS user, we decided that we would have to develop the product differently than we normally do. We would have to go much more slowly and deliberately than usual to ensure that UltraChat had the features sysops and users demanded, while maintaining the emotional "feel" that would be critical to its success with BBS users.

In the end, UltraChat took more than 17 months to develop. At every step of the way, we were paying very close attention to the reaction of sysops and BBS users alike. The design of UltraChat was changed and augmented many times during its development, reflecting what our continuing research showed us about chat.

What you hold in your hands today is the most powerful, most elaborate chat system ever developed for use on a bulletin board system. With UltraChat, you can provide virtually any chat function you can imagine.

Our research has not ended. At eSoft, we are always looking for your feedback and comments about our products. If there are functions you feel UltraChat needs, we want to hear from you. We will be incorporating your ideas and thoughts into an UltraChat upgrade, which will be provided free of charge to registered UltraChat 1.0 sysops. To reach us, post a message on our eSoft Support BBS in the New Features message area, or send us mail to:

New Features
eSoft, Inc.
15200 E. Girard Ave., Suite 3000
Aurora, CO 80014
USA

An UltraChat Orientation

UltraChat is designed with a completely modular approach to configuration. This structure, unlike anything ever designed for chat systems, allows you complete control over the form and function of your chat environment. If you wanted, you could have a user named JOHN DOE receive a totally or partially different configuration if he calls on line 3 between 10:00 and 2:00 on any Friday the 13th. Of course, in most cases, your configuration needs are more modest. UltraChat gives you the control you need and want -- from the practical, to the bizarre, to the downright amusing -- imagine a different configuration only on April Fool's Day!

The capabilities of UltraChat are expansive and powerful. With UltraChat's modular structure, you can take advantage of any or all the features described below.

General Features of UltraChat

- Define up to 65,535 chat channels (conferences) per group, which can be conditioned on privilege level, access flags, line number, modem speed, active group, user name, language selection, time, day, calendar date, and/or a special passed parameter.
- Up to 250 groups can be defined; a total of over 16 million distinct "places to be within chat" can be defined.
- Any chat command or configuration item can be conditioned on privilege level, access flags, line number, modem speed, active group, user name, language selection, time, day, calendar date and/or a special passed parameter.
- All commands, command keystrokes, prompts, text displays, error messages, etc., are 100% configurable. Each user or group of users may have a different combination if you wish.
- Named or numbered channels (conferences) may be used; the two types can be mixed within a single chat environment.
- One private channel per user, and private 1-to-1 chat.
- Moan macros, also known as "action words" in other BBS packages. Allows completely customized "canned chat message

sequences" to be sent by users. For each macro, a text display can be sent to the user sending the macro, the user receiving it, or others in the public channel; each one can be conditioned on whether you direct the macro to a specific user and/or entered a text string with the macro.

- Full support for and enhancement of TBBS billing class timing.
- TDBS API allows smooth interaction between UltraChat and TDBS application programs. (UltraChat is supplied with a TDBS-based survey and matchmaking system.)
- User can monitor a secondary channel; allows eavesdropping of a channel other than the one you are on.
- Users can set a password for private channels, or otherwise control their channel by invite/uninvite commands.
- Who's online displays that offer:
 - All users on your channel.
 - All users on any channel.
 - All users on the system, whether in or out of the chat environment.
 - Three-line bio information for new users, individual users by name, alphabetical by user (forward or reverse).
 - Three-line bio auto-lookup of "From:" user from a TBBS message read prompt.
 - Full control over formatting of the who's online display for each type of who's online command you provide; include or exclude information such as user name or ID, location, conference channel name or number, three-line bio, announce string, and much more.
 - Any who's online display can show or suppress invisible users and/or idle lines.
 - Any who's online display can be conditioned on privilege level, access flags, line number, modem speed, active group, user name, language selection, time, day, calendar date and/or a special passed parameter.

- Chat exit commands include:
 - Exit back to TBBS menu.
 - Exit from private channel to public channel.
 - Perform Type 10 system logoff.
 - Logoff without display (immediate carrier drop).
 - Chat exit can send logged off or dropped carrier message to other users.
 - Chat exit can send left chat message to other users.
 - Chat exit can send left channel for another message to other users.
- UltraChat offers a variety of ways to provide help online:
 - Nine (9) command definable, each with its own help file display.
 - Nineteen (19) commands definable with text string display to implement commands such as version, system name, current date and time, etc.
- A user can send a message to another user in a variety of ways:
 - Send specific message to specific user.
 - Send specific message to all users. (Like all UltraChat command, this may be set to be available or not based on privilege level, access flags, line number, modem speed, active group, user name, language selection, time, day and/or calendar date.)
 - Page user (sends canned message to specific user).
- There are a number of ways to change chat channels or conferences within UltraChat:
 - Go to public channel by number.
 - Go to a public channel by name.
 - Go to a user's private channel (can be conditioned on invitation or password).

- Users can chat privately with one another within the chat environment. Private chat options are:
 - Ask a user to chat and wait for his or her arrival.
 - Ask user to chat with user-entered message and continue where you are until other user accepts; users placed in 1-to-1 chat upon acceptance.
 - Ask user to chat with yes/no invitation; users placed in 1-to-1 chat upon acceptance (canned message delivered on refusal).
 - Ask user to chat with yes/no invitation; users placed in 1-to-1 chat upon acceptance (user-entered message delivered on refusal).
 - Chat demand (chat shanghai) allows a user to “yank” another user into 1-to-1 chat without regard to invitation or acceptance of same.
- Users may configure themselves to be chattable (may be yanked), limited chattable (invitation type only), not chattable (all invitations refused), or time chattable (requests allowed only every “n” minutes).
- UltraChat allows a user to customize his or her own chat preferences. Like all UltraChat commands, you can selectively make these available or unavailable to your users. They include:
 - Three-line bio, allowing user-entered profiles.
 - Chat availability (whether private chatting is desired or not).
 - Ignore/remember for all or specific users (suppresses all chat traffic from user or users).
 - Invite/uninvite for all or specific users (controls whether someone is able to enter your private chat channel or not).
 - Visible/invisible, controls whether others online can see you in who's online displays.
 - Message acceptance level (whether private messages are accepted; whether logon announcements are desired).

- Announce string, a vanity item optionally displayed in who's online displays.
- Name for private chat channel.
- Password for private channel access control.
- User-customized "handle" to override their normal login name or ID.
- Color settings used within chat displays.
- Moan macro enable/disable.
- Shortcut commands required/not required.
- UltraChat also supports moderated conferences. The following features are available; as with all UltraChat features, you can control who has access to each one:
 - Create a new named conference and become the moderator.
 - Moderator can silence a user in the conference.
 - Moderator can force a user in the conference to be logged off.
 - Moderator can force a user to be suspended (marked deleted in the TBBS userlog).
 - Moderator can pass moderator status to another user.
 - Moderator may resign, leaving the conference unmoderated.
 - Moderator may assign a password to a moderated conference.

UltraChat Customization

UltraChat is supplied with several sample configurations to suit a wide range of applications. The sample configurations are designed to emulate the chat facilities of various popular chat-oriented BBS systems, including DLX, Oracomm, and Major BBS. We have also supplied a sample configuration that emulates (with some enhancements) the TBBS internal chat facilities. These sample configurations are designed to be “plug and play” – in other words, you can drop UltraChat and one or more of the sample configurations into your system and be ready to run.

Once you have gained some experience with UltraChat, you may wish to make modifications to the samples we supply and customize them as you see fit. Or, you may wish to create your very own chat system of your own unique design. UltraChat gives you dozens of tools you can use in a variety of ways to achieve these results.

Please note that UltraChat configuration takes a great deal of time and patience to understand fully. We encourage you to experiment with UltraChat configurations, making your test configurations available only to sysops or a small group of trusted users. This allows you to make changes and then test them to determine how and why things work the way they do. Once you have tested your changes thoroughly, you can make them available to all your users.

An Introduction to UltraChat Customization

Customizing UltraChat consists of changing options in the UCHAT.CTL configuration file. This configuration file contains the various commands which tell UltraChat what to do and how to behave for each user.

There are two basic types of configuration in UltraChat: statements and commands. Statements allow you to configure general things within UltraChat, like how many channels are available within the chat environment.

Commands, however, make up the bulk of UltraChat configuration. In UltraChat, command configuration is composed of a simple relationship between the commands themselves and command elements. Commands are things people can do in UltraChat, such as exit back to the BBS or send a private message to someone online.

Commands generally consist of a keystroke or keystroke sequence that a user types in order to activate a particular operation within the UltraChat chat environment. For example, the command `/WHO` might be configured to show a list of the users online at the moment.

Command elements (or just elements for short) are the results or effects of a command, and include responses displayed to the screen, things sent to other users, prompts, and error messages.

Some commands don't have any elements associated with them. An example of such a command might be one to exit the chat facility immediately and return to the BBS. In this case the user might be put back into the BBS without any interim response or prompting of any kind.

Elements come in four forms: responses, prompts, remote displays, and error message displays.

Responses are text shown to the user when a command is issued. For example, if a user changes a chat channel, UltraChat may send a response to acknowledge that the channel was changed and what the new channel is.

Prompts request additional information from a user. For example, if a user issues a command to change a chat channel, but forgets to designate the channel they want to go to, UltraChat may prompt the user to designate the channel number.

Remote displays are text shown to a user (or users) who is the target of a particular command. For example, if a user pages another user online, the remote display is sent to the user who is being paged.

Error message displays, as the name suggests, are text shown to a user when they make a mistake in the formatting or syntax of a command.

In summary, UltraChat configuration involves designating commands and command elements. Commands involve a key or key sequence that is used to activate the command within the chat environment. Command elements are the results of a command, and include responses, prompts, remote displays, and error messages.

How Commands and Elements Are Connected

As mentioned previously, there is a simple relationship between a command and its command elements. But how is this relationship formed inside the configuration file? How do you connect a command to its elements?

Commands and elements are connected through the use of a simple, easy to understand naming convention. We'll explain this with an example: that of changing channels. You may wish to refer to the command reference chapter in this manual to follow along.

Let's take the case of the basic channel change command, CHANNEL#01_CMD. This is channel changing command number 1, designated by the "01" in the command name. These numbers identify variations on the command. For example, CHANNEL#02_CMD is also a channel changing command – but it is used to toggle back and forth between public and private chat channels. The command reference chapter lists and explains all the various commands available to you, organized by functionality.

Back to the CHANNEL#01_CMD command. What about its command elements? What is shown to the user, and how do we associate or connect it to this command?

The "01" portion of the command name comes into play here. For example, if we want to have text that's displayed to a user after they execute the CHANNEL#01_CMD command, we could use the command element CHANNEL#01_CONFIRM. Note that this command element name also contains "01" in its name. This is how a command and its elements are associated.

The CHANNEL#02_CONFIRM element is also a confirmation response, but it would go with the CHANNEL#02_CMD command, since they're both "02" (or the number 2 channel changing function).

We know, therefore, that the following command elements would go with the CHANNEL#01_CMD command:

- CHANNEL#01_CHANPMT
- CHANNEL#01_ENTRY
- CHANNEL#01_EXIT

But that none of these command elements go with CHANNEL#01_CMD at all:

- CHANNEL#02_CHANPMT
- CHANNEL#04_NOTINV
- CHANNEL#06_PWPMT

These command elements don't belong to the CHANNEL#01_CMD, because none of them have "01" in their names. They are therefore associated with or connected to different channel changing commands.

The command reference chapter has a complete list of available command elements, organized in the same groups as the commands they go with.

You should also take the time to review the sample configuration files provided with UltraChat for practical, real-life examples of how commands and commands elements are used in typical UltraChat configurations.

A Palette of Functionality

When you set out to customize UltraChat, you will be specifying commands and command elements for functions you want UltraChat to have. You select the desired commands and elements from a wide variety of those which UltraChat is capable of providing. Like the TBBS menu command types you use inside TBBS menus, UltraChat provides a palette of chat functionality you can pick from and mix in any way you see fit. If you do not wish UltraChat to offer a particular feature, such as the ability for a user to become visible or invisible to other users, then simply omit the commands and functions related to that feature.

Or, if you want the feature, but want it to behave differently than it does in one of the sample configurations provided with UltraChat, then you can add, delete, or modify the commands and elements associated with that command.

Where UltraChat really shines, however, is that you can make available mixtures of configurations, where some features are available to some users, while other features are available only to certain other users. UltraChat configurations can change based on line,

user privileges and access flags, modem speed, which group (usually top menu) is active, user name, language selection, day, date and/or special passed parameters. By using TBBS' own access restrictions at the menu level, you can also make configurations change based on the time of day or day of the week.

If you wish, you can even offer multiple configurations to users and let them pick which they are most comfortable with – and still allow all the users to interact regardless of which configuration they choose. They each see things as they expect, even though other users may be using different emulations.

Tutorial – Recreating TBBS Chat Using UltraChat

To assist you in understanding how to configure UltraChat, we have created a brief tutorial. This tutorial is designed to guide you through the initial steps of setting up UltraChat to operate like TBBS' internal chat facilities. TBBS' chat is relatively simple, and if you've been running TBBS for any time, it's likely a chat system you're familiar with.

This tutorial will not guide you through the entire process of recreating TBBS' chat; rather, we'll cover the most important aspects, with the goal being to orient you to an approach to configuring UltraChat: How it's done, what's involved, how commands and command elements relate, and so on.

eSoft has supplied a pre-made TBBS configuration for UltraChat, which can be found in the \TBBS directory on your UltraChat disk. You may wish to print this out so you can follow along during the tutorial. We do, however, encourage you to perform the steps in the tutorial yourself independent of the TBBS configuration we provided you.

Creating the Configuration File

UltraChat configuration takes place within a text file: UCHAT.CTL. Therefore, the very first step involved in creating a configuration is to use your text editor (or a word processor in non-document mode) to create an ASCII text file by this name.

While in edit mode, you should go ahead and configure the "basics" of UltraChat, such as your name, BBS name, command parsing style, and so on. Substituting the names you wish to use, add the following configuration statements to your newly created UCHAT.CTL file:

```
SYSTEM_NAME_TEXT "The Chat BBS"  
SYSTEM_ID_TEXT "CHATEBS"  
SYSOP_NAME_TEXT "John Doe"  
  
UCHAT_DIR C:\TBBS\UCHATTMP  
  
TBBS_PARSER  
  
MIN_CHANNEL_NUM 1  
MAX_CHANNEL_NUM 40  
LIST_ME_SWITCH NO  
KILL_LINE_KEY ^X
```

The first three statements all define names for you or your system, and should be a part of all UltraChat configurations.

The UCHAT_DIR statement is also a part of all UltraChat configurations, and designates the directory where UltraChat will store temporary information files.

The remaining statements are general config statements which setup the chat environment to be like TBBS. TBBS_PARSER tells UltraChat to accept line numbers (instead of names) in chat commands. The MIN_ and MAX_CHANNEL_NUM statements in this context pertain to CB emulation in TBBS, the only area in TBBS chat where channel numbers are used. In TBBS, they range from 1 to 40. The LIST_ME_SWITCH sets it so that the user doesn't see himself in who's online displays (the TBBS way of doing it), and the KILL_LINE_KEY is the keystroke used to abort input (Ctrl-X in a TBBS chat section).

To know which general configuration statements to use, you should peruse the "General Configuration Statements" section of the Command Reference Chapter. Read the descriptions, decide which apply to you, and place them at the top of your configuration file. The statements we used above are those that would pertain to a TBBS emulation.

Entering Chat

In TBBS, when a user enters chat they are entering a named, public conference. The next step in configuring our emulation is to implement this type of construct in UltraChat.

A review of the general configuration statements in the Command Reference reveals the `DEFINE_CONF` statement. Since this is used to define named conferences, that's what we'll use to create a conference for our TBBS emulation. Add this line to your configuration file:

```
DEFINE_CONF 1 "Public Chat"
```

The 1 is the public conference identifier. We picked 1, since this is the first named conference we'll create. Our name for the conference is "Public Chat," but you can use whatever you wish.

We need to make sure that when we put UltraChat in a TBBS menu entry that the user enters this public channel. To do this, create a TBBS menu entry similar to this one:

```
Entry:  
<E>nter Public Chat  
Key=E Type=203 Opt Data=/E:6,1
```

The `/E:6,#` switch tells UltraChat to enter a named channel on entry. The 1 used with the switch says to enter the named conference 1, which is the identifier we used with our `DEFINE_CONF` statement just a moment ago.

That's all that's needed to get users to enter our named public conference, just as they would with TBBS.

Providing a Way Out

Now that we've given users a way to get into our TBBS-like chat section, we need to give them a way to get back out once they're in. When you're configuring UltraChat fresh (as we are here), this should be the very first command you define, since you'll need it for testing any other commands – unless you want to keep rebooting the PC! If you don't provide a command to exit chat, you'll get in and never be able to get back out again.

In TBBS, there are really two types of exiting: one exits back to TBBS from a public chat section, and the other exits from private chat back to public chat. They both use the same keystroke (which UltraChat supports just fine).

If you peruse the Command Reference chapter, you'll see exit commands that match these two needed exit styles: `EXIT#01_CMD`, which exits back to the BBS, and `EXIT#02_CMD` that exits from private to public chat. (There are other exit styles too – but they wouldn't apply to a TBBS emulation.)

Let's first add the public to BBS exit:

```
EXIT#01_CMD "/Q" MODE=CDEHI PARTIAL EXACT
```

Since this is the first command we've added, some explanation is in order.

Each command begins with the command name, in this case `EXIT#01_CMD`, the command UltraChat provides to allow exit from chat back to the BBS.

The next part of any command is the text which a user types to make it happen. In TBBS chat, an exit is done with the `/Q[UIT]` command, so we'll use just `/Q` here since TBBS can act on the command with as little as that.

The `MODE=` section is a little tough to understand at first, but it's very simple. The modes are a series of letters which designate where within the chat environment this command is active and allowed. The beginning of the Command Reference chapter of the manual lists the available modes and what they mean. The mode letters we've used mean that this command is available anywhere except within private chat (since in private chat in TBBS, the `/Q` command does not exit a user to the BBS).

The `PARTIAL` keyword means that a partial match on the command is allowed, and this is important for TBBS emulation. The reason is that `/Q` is the core of the command, but TBBS chat would allow a user to type `/Q`, `/QUIT`, `/QU`, `/QUI` or even `/QMODEM` and all would have the same effect. By using `PARTIAL` here, it tells UltraChat to allow this leading, partial style of matching.

The EXACT keyword means that parameters are not allowed on the command. For example, /QUIT would be allowed, but /QUIT FOOBAR wouldn't be allowed. EXACT means that only the command can be entered – no parameters are acceptable.

That's it! Our first command is done. To test what we've done so far, from DOS, type:

```
UCHAT COMPILE
```

IMPORTANT: Make sure you recompile every time you make a change to configuration. If you don't, your changes will not take effect!

This compiles the configuration information so TBBS can use it. Run TBBS, logon, and access UltraChat. When you enter chat, at this point, you won't see anything. But you should be able to type /Q, press Enter, and exit back to the BBS. If you've followed the tutorial exactly so far, this should work perfectly.

Exiting From Private to Public Chat

As mentioned in the previous section, TBBS uses the /Q command for another purpose too: to exit private chat and go to public chat. That means we have another exit command to define.

The command that allows this type of exit is the EXIT#02_CMD command. You can determine this by perusing the Command Reference chapter. So let's add this command to our configuration:

```
EXIT#02_CMD "/Q" MODE=B PARTIAL EXACT
```

This is almost identical to the exit command we configured in the previous section. There are two differences, however.

First is the command name. In this case, we use EXIT#02_CMD since it's the command we're wanting to happen in this situation.

Secondly, the MODE = section is different this time. As we mentioned before, the MODE = controls where within the chat environment that this command can be used. We're using MODE = B in this case, since it will limit the applicability of this command to private chat only – which is the only place we want it to be active.

Again, the beginning of the Command Reference section of the manual explains the meaning of the various **MODE =** letters.

That's it!

Let's summarize where we're at:

- We have two **/Q** commands, both of which perform a type of exit.
- One of the **/Q** commands exits from public chat to the BBS.
- One of the **/Q** commands exits from private chat to public chat.

Go ahead and recompile the configuration (from DOS, type **UCHAT COMPILE**, as we did before), but you won't be able to test the second type of exit since we haven't given a way to enter private chat quite yet. That will come later on.

Listing Who's Online

We're coming right along with our new chat system. One of the most popular and necessary features of chat is the who's online display. Let's add a way for our users to list who is online.

In TBBS, if you study the who's online displays it provides, you'll find that it provides three versions:

- Who's online, system-wide. This is the **/W[HO],S[YSTEM]** (or **/W,S** for short) command, and is available anywhere within TBBS chat.
- Who's online in this chat section. This is the **/W[HO]** command, and it lists who is in the chat area with you.
- Who's online in this chat section when nobody else is there. This is also the **/W** command, but when you issue that command when nobody else is in the chat section with you, it changes behavior to a **/W,S** command automatically.

We can emulate all three of these commands. Let's begin with the last of the three, the **/W** command behavior when nobody is in chat with you:

```
WHO#01_CMD "/W" MODE=CH PARTIAL EXACT
WHO#01_HEADER "No other callers in
conference: %CHANNEL%^M^M
+ Callers currently on system%^M^M"
WHO#01_NORMAL "Line %RLINE:+02%: %RUSER%;
%RLOCATION% (%RCOMMAND%)^M"
```

UltraChat lets you define several who's online commands. We picked the WHO#01_CMD since it's the basic type of who's online display, and this is the first one we're creating. The MODE=CH means that this command is active only within public channels when nobody else is present. Refer to the description of the MODE= parameter in the Command Reference chapter. PARTIAL means partial matching is enabled, so /WHO is allowed too. And EXACT means no parameters are allowed on the /W command.

The command elements we used are patterned after TBBS' chat displays.

Next, let's add the /W,S command:

```
WHO#02_CMD "/W" MODE=ABCDHI PARTIAL BLANK
WHO#02_HEADER "Callers currently on sys-
tem%^M^M"
WHO#02_NORMAL =WHO#01_NORMAL
```

That's it! In this case, we picked WHO#02_CMD since this is the second of the basic who commands we're using. The MODE= here allows the command to be active everywhere, PARTIAL allows partial matching.

The BLANK parameter here means that a blank must follow the command. "Blank" means a blank (a space), a comma, (,) or a tab must be typed after the command itself. Since the command we're emulating is /W,S, this means that the user must, at the minimum, type a comma after the /W for UltraChat to know it matches this particular who's online command, which makes the behavior TBBS-like.

NOTE: Notice how the WHO#02_NORMAL command element has "=WHO#01_NORMAL" as its text. This is called command element sharing, and is a way to save time and UltraChat memory space by reusing previously defined command elements. In this case, the command element WHO#02_NORMAL will use the

same text as WHO#01_NORMAL. This is designated using the equal sign (=) followed immediately by the command element text to share. IMPORTANT: the shared element must be already defined previously in the file, as would be the case here, in order to be shared.

Lastly, let's add our who's on the channel command:

```
WHO#07_CMD "/W" MODE=BDI PARTIAL EXACT
WHO#07_HEADER "Callers in conference:
%CHANNEL%M^M"
WHO#07_NORMAL =WHO#01_NORMAL
```

There! The WHO#07_CMD is the command provided specifically to list who is in the same channel as you (the user). The MODE= command limits this command to places where there are others with you so it has the desired effect in the desired places.

Spend some time in the who's online display section of the command reference chapter to acquaint yourself with the commands and elements available to you.

Letting Users Communicate Privately

Another popular command you'll probably want to implement is a way for users to send private messages to other users. In TBBS chat, this is the /S (send) command. UltraChat allows you to configure several such commands. We'll only need one, however, so go ahead and add that command now:

```
PRIVATE#01_CMD "/S" MODE=BCDHI PARTIAL
```

Our MODE= portion allows the command to be used anywhere within the chat environment, and PARTIAL allows partial matching to emulate TBBS chat operation.

We used PRIVATE#01_CMD, but PRIVATE#01_CMD through PRIVATE#09_CMD are all the same (we could have picked any one of them).

We will, of course, need a few command elements to make this command behave like we want. In the section of the Command Reference chapter that discusses private message sending, you'll

find several command elements available to you. Peruse the ones available to find the ones you want to use. For TBBS emulation, we'll use two of them.

The first is a confirmation message shown to the user who issued the command. Go ahead and add this now, immediately following the command we just put in:

```
PRIVATE#01_CONFIRM "Message sent to line
%RLINE%^M"
```

This is identical to the prompt normally displayed by TBBS chat when you send a private message. The %RLINE% insertion parameter is used here, because it refers to the remote users line number (the line of the user we just sent the private message to). Note the ^M at the end – this is a carriage return. Remember that you must put these in explicitly whenever you want a carriage return to be displayed.

Now that UltraChat is confirming our message sending, let's add a way for the remote user to see the message:

```
PRIVATE#01_REMOTE " ^M^M(%RLINE:+02%:%RUSER%,
Pvt Msg) %MSG%^M"
```

This would result in a display that might look something like this:

```
(02:ALAN, Pvt Msg) Hello there!
```

Very TBBS-like! The text begins with a double carriage return, as does TBBS when it displays an incoming private message. The %RLINE:+02% insertion parameter gives the line number in a TBBS style. The %RUSER% refers to the user entering the command. Finally, %MSG% shows the message the person who issued the command typed in.

You'll also notice some punctuation and such to make the display look TBBS-like.

That completes the private message send command. It now allows a user to type in a send command just as they would in TBBS.

Compile the configuration, logon, and test at this point if you'd like.

Summary

This is as far as we'll take you in this tutorial. After completing it, you should have an understanding of the following:

- The relationship between commands and command elements.
- Commands and command elements are listed in the Command Reference chapter. You should peruse this chapter to learn about the capabilities of UltraChat, and to make changes to existing configurations or make new ones.
- The `MODE =` command parameter is used to indicate where and under what conditions within the chat environment that a command will be active.
- The meaning of the `PARTIAL`, `EXACT` and `BLANK` keywords. (These are also explained in the Command Reference section of the manual.)
- Command keystrokes can be duplicated, with the same command keystroke executing one command in some sections, and other commands in other sections. (The `/W` who's online command we configured above is a perfect example of this concept.)
- Command elements can be repeated by using command element sharing. (This also saves memory.)

If you're interesting in continuing with the TBBS chat emulation, you can refer to the sample configuration file supplied with UltraChat. (You may notice some slight differences between the one supplied and the exercises in this tutorial; the one supplied on disk is more verbose and complete.)

Optimizing UltraChat Performance

Internally, UltraChat is a very complex product. Under the hood where you and your users cannot see it, UltraChat is actually performing hundreds of operations, incorporating a significant amount of processing. As such, there are a number of ways you can optimize the performance of UltraChat. Here are some hints for optimization:

- The UCHAT_DIR directory should be placed in a RAM disk drive, as it gets accessed frequently by UltraChat.

CAUTION: RAM disks can often conflict with TBBS memory segments. Use a RAM disk only on a 386 (or greater) processor, and make certain that the RAM used by the RAM disk is located in extended (also known as XMS) memory. DO NOT place the RAM disk in expanded (EMS) memory!

IMPORTANT: The RAM disk drive must be referenced in your TBSPATH= statement in the RUNBBS.BAT file in order for TBBS and UltraChat to look for files there. Refer to your TBBS manual for details on setting the TBSPATH. For best performance, put the RAM disk first sequentially in the TBSPATH= setting, before any other directories.

The UCHAT.\$\$\$ and UCHATMON.\$\$\$ files should be copied to the RAM disk (see above) and removed from the TBBS directory when you start-up your system. This can be incorporated into your RUNBBS.BAT file. Since these files too are accessed frequently, having them in a RAM disk will improve performance.

CAUTION: These two files are the compiled version of your UltraChat control files. If there are absent when you run the system, UltraChat will not function. Make sure you execute the UCHAT_COMPILE command prior to copying them from the TBBS directory to the RAM disk drive. Once they are copied to the RAM disk, delete them from the TBBS directory, since UltraChat will look there first before accessing the RAM disk.

- UCHAT.CTL should be kept as small as realistically possible. Running more emulations, or running bigger emulations, causes the control file to be much larger. The larger it is, the longer it takes UltraChat to process the information in it, both during compilation and later online.

- Whenever possible, use the command element sharing technique discussed at the beginning of the Command Reference section of the manual. This will save space in the internal tables of UltraChat, reducing memory and processing overhead.
- As with TBBS itself, disk caching software – when carefully installed – can be of great benefit to the system. Use a solid, reliable disk cache product. Make sure you disable any type of write caching, and make sure the cache memory is located in extended (XMS) memory and NOT in expanded (EMS) memory. Use a cache only on a 386 (or greater) machine. Disable all advanced caching options available in your cache software to reduce the possibility for conflicts and other problems that can arise from not using simple, basic caching options.

BEST OPTION: Use a caching disk controller instead of resident software based disk cache. This eliminates the possibility of memory conflicts with TBBS.

UltraChat Configuration Reference

This section contains a complete configuration reference for UltraChat. Its purpose is to serve as a comprehensive resource when customizing UltraChat.

IMPORTANT NOTE: Please refer to the sample configurations supplied with UltraChat for examples of configuration usage. Specific examples of each statement, command, or command element are not provided in this reference, but rather, within the sample configuration files.

The Parts of UltraChat Configuration

Configuration verbs in UltraChat come in three forms: statements, commands, and command elements.

Statements are for the benefit of UltraChat itself, and allow you to control broad, sweeping functionality. For example, there is a configuration statement that allows you to set how many channels are available within the chat environment.

Commands are for the benefit of people using UltraChat, and form the basis of most of the configuration you do for UltraChat. Commands designate what functions are available to users while within the chat environment. For example, there are several different commands that allow you to create who's online displays.

Finally, command elements relate to specific commands, and tailor what each command does. Command elements always designate text that is shown to users as a result of a command. For example, a channel changing command might display confirmation text to a user. Command elements come in four types: responses (text shown to the user issuing a command), prompts (text shown to the user issuing the command asking for more information), remote displays (text shown to other users as the result of one user performing a command), and error messages (shown to a user issuing a command when they make a mistake).

Command Formatting

Since commands form the basis of most UltraChat configuration, it's important to understand how commands are formatted when you configure them. Command verbs always have the suffix `_CMD`, such as `EXIT#02_CMD` and `WHO#07_CMD`. All commands adhere to this format:

```
<cmd_type> <word> MODE=<mode> [PARTIAL]  
[EXACT] [BLANK]
```

For example:

```
EXIT#01_CMD EXIT MODE=BCD EXACT
```

This defines a command string "EXIT" which when typed by a user will cause an `EXIT#01_CMD` type of command to be executed as long as `EXIT` is the only thing the user typed on the line. The parameters are as follows:

<cmd_type> – one of the listed command types.

<word> – The "word" to match to cause execution of the command. The word can be up to 14 characters, and can contain embedded spaces if the word is put in quotes.

MODE=<mode> – The "modes" when the command is valid. **<mode>** can be any string of the letters A through I. The letters **MUST** be together and **MUST NOT** be separated by spaces or any other characters. The allowable mode characters are:

- A – called from a TBBS menu as a one-shot
- B – valid within one-to-one style private chat
- C – valid within numbered public chat channel when no one else is present
- D – valid within numbered public chat channel when others are present
- E – valid when called from the TBBS read message prompt
- F – valid within a private channel when no one else is present
- G – valid within a private channel when others are present

- H – valid within a named conference when no one else is present
- I – valid within a named conference when others are present

There are also three keywords that can be used with each command definition:

PARTIAL – When this command is matched, ignore the rest of the input up to a blank, tab, comma or end of input line. This can be used to make /W, /WHO, /WHAT, /WHOCARES all be acted upon as though only /W was typed. This is necessary for TBBS internal chat compatibility.

EXACT – The user is not allowed to add any parameters to the command, or it will not be matched as being this command. This can be used in conjunction with PARTIAL to make a command that can match loosely, but can't have parameters (the /QUIT command in TBBS is an example of this).

BLANK – The user **MUST** follow the command with blank space. This can be a blank, tab or comma (comma is compatible with TBBS built-in chat). This can be used if one command is a subset of a moan macro word.

IMPORTANT NOTE: COMMANDS ARE SEARCHED SEQUENTIALLY, SO MAKE SURE THAT IF TWO COMMANDS START WITH THE SAME LETTERS, THE LONGER ONE IS LISTED FIRST. OTHERWISE THE SHORTER ONE WILL MATCH AND BE USED!

Command Element Formatting

Command elements always follow a command within the UltraChat configuration. Command elements are linked to commands by a shared prefix. For example, a command element named INTERACT#02_ON is linked to the command INTERACT#02_CMD, because they share the same prefix: INTERACT#02.

Command elements usually have a string of text as a parameter. Since this is the case, except where noted, command elements can be assumed to have the following format:

<element> <text>

Here's an example of a command element configuration:

```
ACTION#01_CONFIRM "-- Generic action --^M"
```

In this example, "ACTION#01_CONFIRM" is the command element, and the text that follows, in quotation marks, is the text that goes with the element.

Command element text can be handled a variety of ways. Refer to the section "Using Text in UltraChat Command Elements" for more details.

Command Element Sharing

Since command element text is often the same for multiple commands, you can save time, memory usage, and configuration file space by using command element sharing. This technique allows you to specify command element text once, then use the same text repeatedly in subsequently listed command elements. Here's an example of this technique:

```
CHANNEL#02_CMD JOIN      MODE=CDFG      EXACT ;JOIN
CHANNEL#02_CMD /J        MODE=BCDFG      EXACT ;/J
CHANNEL#02_CHANPMT "Who's private channel would
you like to join? "
CHANNEL#02_NOTINV "Sorry, but you're not invited
to %RUSER:C%'s
+ | private channel.^M"
CHANNEL#02_ENTRY "%RUSER:C% has just arrived
from another
+ | channel.^M"
CHANNEL#02_EXIT "%RUSER:C% has just gone to
another channel.^M"

CHANNEL#03_CMD JOIN      MODE=CDFG      ;JOIN <who>
CHANNEL#03_CMD /J        MODE=BCDFG      BLANK ;/J <who>
CHANNEL#03_CHANPMT =CHANNEL#02_CHANPMT
CHANNEL#03_NOTINV =CHANNEL#02_NOTINV
CHANNEL#03_ENTRY =CHANNEL#02_ENTRY
CHANNEL#03_EXIT =CHANNEL#02_EXIT
```

In this example, the command elements are defined once in conjunction with the CHANNEL#02_CMD command. (The com-

mand elements here are CHANNEL#02_CHANPMT, ..._NOTINV, ..._ENTRY, and ..._EXIT.) Then, the command elements of the CHANNEL#03_CMD command share the previously defined elements by using an equal sign (=) followed by the element to share, such as:

```
CHANNEL#03_ENTRY    =CHANNEL#02_ENTRY
```

What this means is that the CHANNEL#03_ENTRY element is to have text identical to the CHANNEL#02_ENTRY element.

The most important reason to use command element sharing is **memory efficiency**. UltraChat has a limited amount of space for configuration information. Each time you uniquely configure a command element, the text of the element consumes more memory. When you use command element sharing, however, two (or more) elements can use the same text, which consumes memory once only. Use command element sharing whenever you don't require separate text for multiple, similar command elements.

IMPORTANT: WHEN USING COMMAND ELEMENT SHARING, THE SHARED TEXT MUST BE DEFINED EARLIER IN THE CONFIGURATION FILE – YOU CANNOT SHARE AN ELEMENT WHICH IS NOT PREVIOUSLY DEFINED!

General Configuration Statements

The general, overall configuration of UltraChat behavior is done with a variety of items called **general configuration statements**. These statements allow you to control a variety of items, such as the directory UltraChat uses, how many channels are available, the length of the inactivity timeout, and much more.

UCHAT DIR <directory>

UltraChat temp directory. This statement allows you to designate the drive and path of UltraChat temporary directory. To optimize the use of memory, UltraChat often temporarily stores information (such as the private message queue) to disk. This directory is used

for that storage. Using a RAM disk for this directory will help increase overall UltraChat performance. Refer to the "Optimizing UltraChat Performance" section.

INCLUDE <filename>

This statement allows you to designate the name of other files that contain UltraChat configuration information, and tells UltraChat to bring the configuration information in those files into effect. If there are multiple configuration files you wish to include, you must use a separate INCLUDE statement for each one. You may nest INCLUDE statements (INCLUDE statements within an INCLUDED file) up to five (5) levels deep.

NEW USERS NUM <number>

Designate the last <number> users as new. This statement allows you designate how many users get displayed when an UltraChat user does a WHO#06_CMD command (discussed later), which allows a listing of "new" users. By setting a value in this statement, you're telling UltraChat to show the last <number> users in the TBBS userlog file that are not marked invisible or deleted when a user issues the WHO#06_CMD command. The UltraChat default is 10 users.

ANNOUNCE CHARS NUM <number>

Size of announce string. This statement allows you to define how many characters maximum are allowed in an "announce" string. The default is 30, the maximum allowed value is 80. Refer to the glossary for an explanation of announce strings. Announce strings are set by a user with the EDIT#01_CMD command, and are displayed online with the %ANNOUNCE% and %RANDOMOUNCE% insertion parameters.

MIN CHANNEL NUM <number>

Set lowest allowed channel number. This statement allows you to configure the lowest channel number allowed in the chat environment. The default is 1, which is the same as Oracomm. (Major BBS, for example, has a minimum channel number of 0.) The minimum acceptable value is 0.

MAX CHANNEL NUM <number>

Set highest allowed channel number. This statement allows you to configure the highest channel number allowed in the chat environment. The default is 9, which is the same as Oracomm. (Major BBS, for example, uses 65535.) The maximum acceptable value is 65535.

PUB CHANNEL NUM <number>

Set highest allowed public channel number. This statement allows you to configure the highest channel number allowed within public chat. The default is that all channels are public. If this number is lower than that set with the MAX_CHANNEL_NUM statement, then all higher numbered channels allow anyone into them, but do not list the channel number on who's online displays. This effectively makes users in those channel "hard to find."

DEFINE CONF <number> <name>

Define a named conference. This statement allows you to define a named chat conference. The <number> parameter designates a numeric conference identifier. **IMPORTANT:** These numbers are not the same as chat channels! The <name> parameter is a string of not more than 30 characters of text. If embedded spaces are desired, place the name within quotation marks.

LAST LINE NUM <number>

Set highest line used on who's online displays and for lookups. This statement allows you to configure the highest TBBS line number that UltraChat will look at when looking up users and doing who's online displays, etc. This is useful if you have some ghost lines that you don't want to have counted. The maximum number is the number of lines your TBBS is configured to support. Who's online display commands, WHO#xx_CMD, use this to limit their scope. This also limits the lines looked at to find users that are targets of commands.

PUBLIC ENTRY KEY <key>

Set the key which allows entry to chat when entering in Oracomm mode. This statement allows you to configure the keystroke that a user must press to enter public chat when they enter UltraChat in Oracomm mode. (The entry mode is designated with the /E: Opt Data switch.) The <key> parameter designates the key. The default key is C.

PUBLIC EXIT KEY <key>

Set the key which allows exit from Oracomm style chat. This statement allows you to configure the keystroke that a user must press to exit public chat when they are operating in Oracomm mode. (The operating mode is designated with the /E: Opt Data switch.) The <key> parameter designates the key. The default key is X.

STOP KEY <key>

Set the key required to stop scrolling display. This statement allows you to designate the key that a user must press to stop a scrolling display where applicable. The default key is S.

PAUSE KEY <key>

Set the key required to pause scrolling display. This statement allows you to designate the key that a user must press to pause a scrolling display where applicable. The default key is P.

YES KEY <key>

Set the key required to answer YES to yes/no prompts. This statement allows you to designate the key that a user must press to answer yes to a yes/no style prompt within the chat environment. The default key is Y.

NO KEY <key>

Set the key required to answer NO to yes/no prompts. This statement allows you to designate the key that a user must press to

answer no to a yes/no style prompt within the chat environment. The default key is N.

TBBS_PARSER

Perform TBBS-style command parsing. This statement, when present, tells UltraChat to do command input parsing in a fashion like TBBS' internal chat. This means that commands that ordinarily accept a user name or ID will instead be looking for TBBS line numbers, like the TBBS internal chat system does.

NAME_NUM_PARSER

Perform mixed style command parsing. This statement, when present, tells UltraChat to do mixed style command input parsing. With this statement active, when UltraChat accepts a command from a user, it will first attempt to match user-directed commands by a name or user ID. If no match is found, it will attempt to use the supplied "name" as a TBBS line number, like the TBBS internal chat system does. This statement is especially useful in situations where you are converting from a TBBS internal chat system, and wish to make the transition easier for users familiar with TBBS chat and its line number oriented command convention.

SHORTCUT_KEY <char>

Designate key used to preface shortcut commands. This statement sets the single character designated by the <char> parameter as the shortcut key for commands. When in shortcut only mode (discussed later), and UltraChat receives a command from the user that begins with this character, it will attempt to match the input against a shortcut command. If no match is found, the input is treated as a public chat message. When UltraChat is not in shortcut only mode, and UltraChat receives a command from the user that begins with this character, it will attempt to match the input against a shortcut command. But if no match is found, an error message is generated.

The TOGGLE#04_CMD, ON#04_CMD, and OFF#04_CMD commands control whether a user is in shortcut only mode or not.

NOTE: While in one-to-one private chat, lines starting with this key are not sent to the user being chatted with, but rather, the line is treated as a command when the Enter key is pressed.

KILL LINE KEY <char>

Designate line input kill (abort) key. This statement allows you to set a single keystroke that will abort or kill a user's command input midstream. When a user in chat is midway through typing an UltraChat command, and they keystroke is issued by a user, their pending command is discarded by UltraChat and the text given in the MSC#09_TEXT configuration statement is printed to the user's screen (to notify them that their input was discarded, the command aborted, etc.).

OVERALL TIMER <minutes>

Set overall chat time limit. This statement allows you to set an overall chat timer in minutes, thereby controlling how long a user can stay inside the chat environment at one sitting. Note that this does not time by login or session, but rather, it times from the moment the user enters chat (public or private). The user is given one minute and two minute warnings when this timer is about to expire as well as another warning when they are removed from UltraChat; the warnings are configured using the MSC#12_TEXT, MSC#13_TEXT, and MSC#14_TEXT configuration statements. The default is no limit.

INACTIVITY TIMER <minutes>

Set inactivity timeout for chat. This statement allows you to set an inactivity timeout for use within the chat environment in minutes. If a user is in chat for the designated number of minutes and does not type anything, they will be logged off the system. This is independent of the TBBS master inactivity timeout (which does not apply when a user is within the chat environment). The user is given two minute and one minute warnings when the timeout is about to expire, as well as another warning when they are logged off the system; the warnings are configured using the MSC#15_TEXT, MSC#16_TEXT and MSC#17_TEST configuration statements. The default is to allow a user to stay inactive until their OVER-

ALL_TIMER, TBBS billing class timer, or overall TBBS login time limit expires.

ENTER STRING <text>

Set carriage return text substitution. This statement allows you to designate a text string that is sent by UltraChat whenever a user in chat presses carriage return (Enter) only without any other text. When this occurs, the string designated by the <text> parameter is sent. If a command matches <text>, that command will be executed instead. HLNT; If you want to have a command associated with the carriage return key (which is the case with Major BBS emulation), then begin the <text> parameter with the KILL_LINE_KEY character since that will make a command a user cannot enter any other way. See the Major BBS emulation configuration file for an example of this technique.

HOT KEYS <mode>

Set private message queue printing mode. This statement allows you to define how UltraChat should do printing of the private message queue when the system has messages waiting (queued) for a user. (Private messages can queue up when a user is not within a window of opportunity when a private message is received.) The <mode> parameter can be one of the following:

FLUSH. This flushes all typeahead hot-keys when the prompt for "press any key" is displayed.

NO_DISPLAY. This makes the private queue not display when there is typeahead in the input buffer. If hot-keys are pressed while the private message queue is being displayed, they are discarded.

Note that the term "typeahead" refers to keystrokes made by a user that have not yet been executed by TBBS, i.e., keys that are pending or typed ahead, hence the term typeahead.

MATCH ONLINE FLAG <a flags>

Establish user-to-user matching for chat and who's online displays. This statement allows you to configure a system of access flag matching for users to be able to interact within the chat environment. If you establish this matching, and users do not match one another, then they will not see each other in the chat environment, and will not be able to interact with one another. In much the same way as TBBS menus, the matching is based on an access flag "mask." There are, however, six characters allowed instead of the usual three. The - (dash), . (period), and X (letter X) characters all mean the same as they do in menus, except now they mean the OTHER user has to have those flags set. The additional characters available are:

M means this flag for the OTHER user must match ME EXACTLY.

O means ANY ONE of the flags designated by "O" in the respective access flag group (A1, A2, A3 or A4) for the OTHER user must match ME.

A means ANY ONE of the flags designated by "A" for the OTHER user must match ME, but all 32 access flags are taken into consideration (i.e., it spans A1, A2, A3 and A4 flag groups).

For example:

```
MATCH_ONLINE_FLAG A1 = AXMMM00. A2 = A-----  
A3 = A----- A4 = A-----
```

In this example, the other user must have A1[2] = X and A1[8] = . or we don't go any further. The other user must also match my A1[3], A1[4], and A1[5] EXACTLY. The other user must also have the same A1[6] -OR- A1[7] as me. Lastly, the other user must have one of A1[1], A2[1], A3[1] -OR- A4[1] set the same as me.

MATCH USERLOG FLAG <a flags>

Establish user-to-user matching for userlog lookup related functions. This operates identically to the MATCH_ONLINE_FLAG statement, except that it covers userlog lookups such as are done in various who's online commands, three-line bio lookups, and so on; performed by a user with the WHO#xx_CMD commands.

LIST ME SWITCH ON/OFF

Designate whether the user's own line is listed in who's online displays. If the ON parameter is used, then the user's line is included in such displays. If the OFF parameter is used, then his own line is not included. The default is OFF.

LIST PRIVATE SWITCH ON/OFF

Designate how private chat is designated in who's online displays. If the ON parameter is used, then all users in private chat will have their channels listed as "PRI:n" where "n" is the person they are in chat with. If the OFF parameter is used, those users will show just "PRIVATE" instead of who they are in chat with. The default is ON.

LIST INVISIBLE SWITCH ON/OFF

Designate whether invisible users are listed in who's online displays. If the ON parameter is used, then users who are marked invisible will be listed in who's online displays using the appropriate command elements. If the OFF parameter is used, then users who are invisible will not be shown in who's online displays and will not be included in totals. The default is ON.

READONLY SWITCH ON/OFF

Designate whether user is in read only mode in public areas. If the ON parameter is used, then the user is in read only mode whenever in a public chat channel or conference, and cannot enter any message traffic into the conference. If the OFF parameter is used, then the user can interact in public areas normally. The default is OFF. NOTE: You will most likely want to use this statement in a RESTRICT block to limit its applicability to the desired grouping or class of users.

Text Configuration Statements

The following configuration statements, like the ones in the previous section, are general in their nature, but all allow you to configure text that is used in various places within the chat environment.

SYSTEM NAME TEXT <text>

This statement allows you to configure the name of your BBS system, and is displayed by the %SYSTEM% insertion parameter in online displays.

SYSTEM ID TEXT <id>

This statement allows you to configure an ID of 8 characters or less, a short version of the name. This is displayed by the %SYSTEMID% insertion parameter, and is also used internally by Ultra-Chat to define your system name.

SYSOP NAME TEXT <name>

This statement allows you to configure the name of the sysop (usually your user name online). It is displayed by the %SYSOP% insertion parameter in online displays.

WAITING TEXT <string>

Set text displayed for %WAITING% insertion parameter. This text is used in conjunction with the %WAITING% insertion parameter, which can be used to show a user when they have messages waiting in TBBS. See also the %WAITING% insertion parameter for full details.

IS TEXT <string>

Set text for "is" portion of %IS/ARE% insertion parameter display. This statement allows you to set the text used when the %IS/ARE% insertion parameter shows the word "is" (or whatever you set) to a user. See also the %IS/ARE% insertion parameter for full details.

ARE TEXT <string>

Set text for “are” portion of %IS/ARE% insertion parameter display. This statement allows you to set the text used when the %IS/ARE% insertion parameter shows the word “are” (or whatever you set) to a user. See also the %IS/ARE% insertion parameter for full details.

UNLIMITED TEXT <string>

Set text shown by %TIMELEFT% when a user has unlimited time. This statement allows you to set the text verbiage shown by the %TIMELEFT% insertion parameter when a user has unlimited time left. See also the %TIMELEFT% insertion parameter for full details.

IGNORED TEXT <string>

Set text shown by the %IGNORED% insertion parameter. This statement allows you to set the text verbiage shown by the %IGNORED% insertion parameter is the user being referred to is being ignored. This is primarily used for who’s online displays. It does not show whether the user issuing the command is being ignored, but rather, it is from the perspective of the person that issued the command. See also the %IGNORED% insertion parameter for details.

INVITED TEXT <string>

Set text shown by the %INVITED% insertion parameter. This statement allows you to set the text verbiage shown by the %INVITED% insertion parameter is the user being referred to has been invited. This is primarily used for who’s online displays. It does not show whether the user issuing the command has been invited, but rather, it is from the perspective of the person that issued the command. See also the %INVITED% insertion parameter for details.

User Related Text Configuration Statements

The following statements allow you to configure certain user-related text strings:

USER#01 TEXT <text>

Set string printed by %FLAGS%/RFLAGS% when TBBS (F) flag is set. This text is displayed as all or part of the expansion of the %FLAGS% or %RFLAGS% insertion parameter, when the affected user has their (F) userlog flag set in UEDIT. Normally used as part of who's online commands for customized status displays.

USER#02 TEXT <text>

Set string printed by %FLAGS%/RFLAGS% when TBBS (R) flag is set. This text is displayed as all or part of the expansion of the %FLAGS% or %RFLAGS% insertion parameter, when the affected user has their (R) userlog flag set in UEDIT. Normally used as part of who's online commands for customized status displays.

USER#03 TEXT <text>

Set string printed by %FLAGS%/RFLAGS% when TBBS (K) flag is set. This text is displayed as all or part of the expansion of the %FLAGS% or %RFLAGS% insertion parameter, when the affected user has their (K) userlog flag set in UEDIT. Normally used as part of who's online commands for customized status displays.

USER#04 TEXT <text>

Set %RANNOUNCE% string. This statement allows you to set the text displayed by the %RANNOUNCE% string when the affected user is currently in "no chat" mode. This can then be used with who's online display to alert others that the user in question is not currently available for chat.

General Chat Text Configuration Statements

The following configuration statement allow you to define various text strings used in a variety of areas of the chat environment:

CHAT#01 TEXT <text>

Public message format. This statement lets you define the visual appearance of public chat message if you want something other than the default, which is to display the message as-is without formatting. It allows you to define prefix information, etc.

CHAT#02 TEXT <text>

People waiting to chat, line-by-line display. Applies only when a user enters chat in Oracomm mode (/E:1 Opt Data switch). This text string is printed once for each user (TBBS line) that is waiting to chat with a person when they enter the chat environment. See also CHAT#03_TEXT, below.

CHAT#03 TEXT <text>

Header shown on entry to chat when people are waiting to chat. Applies only when a user enters chat in Oracomm mode (/E:1 Opt Data switch). This text serves as a header for the CHAT#02_TEXT displays (mentioned previously). Upon entry to chat, this text is shown to the user if there are other users waiting to chat. This display is then followed by one or more CHAT#02_TEXT displays (see above) to form a coherent display.

CHAT#04 TEXT <text>

Prompt whether to chat with a user who is waiting. Applies only when a user enters chat in Oracomm mode (/E:1 Opt Data switch). This prompt text is shown to a user when one other user is waiting to chat with them when they enter the chat environment. After the user is listed (with the CHAT#02_TEXT and CHAT#03_TEXT, discussed above), this prompt line is displayed. It serves to ask the user whether he wants to chat with the one single user who is waiting to chat.

CHAT#05 TEXT <text>

Prompt whether to chat with one of the users who is waiting. Applies only when a user enters chat in Oracomm mode (/E:1 Opt Data switch). This prompt text is shown to a user when more than one user is waiting to chat with them when they enter the chat environment. After the users are listed (with the CHAT#02_TEXT and CHAT#03_TEXT, discussed above), this prompt line is displayed. It serves to ask the user whether he wants to chat with one of the users who is waiting to chat.

CHAT#06 TEXT <text>

Header shown on entry to chat when people are waiting to chat. Applies only when a user enters chat in Oracomm mode (/E:1 Opt Data switch). This text serves the same purpose as the CHAT#03_TEXT display (mentioned previously), in that it is displayed when the user enters the chat environment. However, this is the text displayed when nobody is waiting to chat with the user, whereas CHAT#03_TEXT is displayed when there are users waiting to chat.

CHAT#07 TEXT <text>

Prompt user whether to wait for another user to come to chat. Applies only when a user enters chat in "wait mode" (either the /E:1 switch, or from a CHAT#01_CMD command). Various chat functions executed from this chat entry method may prompt a user whether they wish to wait for another user to come to chat. This is the text shown to prompt the user to make this choice. When they answer yes, they will be placed into a wait mode for the other user to come to chat; if they answer no, the command will be aborted. This prompt asks the user which behavior they want.

CHAT#08 TEXT <text>

Waiting for chat to begin message. Applies only when a user enters chat in "wait mode" (either the /E:1 switch, or from a CHAT#01_CMD command). Related to the CHAT#07_TEXT discussed above, this text is shown to the user when they choose to wait for another user to come to chat.

CHAT#09 TEXT <text>

Tells of the other user leaving private chat. This is related to the CHAT#07_TEXT, CHAT#08_TEXT and their related commands. Once a user decides to wait on another user, and the other user does come to chat, this message is displayed when the other user leaves when the private chat is completed.

CHAT#10 TEXT <text>

Tells of the other user entering private chat. This is related to the CHAT#07_TEXT, CHAT#08_TEXT, CHAT#09_TEXT and their related commands. Once a user decides to wait on another user, this text is displayed when the user they are waiting on comes into the chat environment, alerting the waiting user that chat can now begin.

CHAT#11 TEXT <text>

Now in public chat alert. Applies only when a user enters chat in Oraconn mode (/E:1 Opt Data switch). This text is shown to the user when they choose to enter public chat from this type of UltraChat entry.

CHAT#12 TEXT <text>

Prompt for use in private one-on-one chat. This prompt is displayed when UltraChat is waiting for input from a user while that user is in private chat with another user. If this text is not used, then nothing is displayed as a prompt in private chat.

CHAT#13 TEXT <text>

Prompt for use in public chat. This prompt is displayed when UltraChat is waiting for input from a user while that user is in public chat with another user. If this text is not used, then nothing is displayed as a prompt in public chat.

CHAT#14 TEXT <text>

User joining channel display. This text is shown to other users on a chat channel when another user enters the channel from someplace other than another chat channel.

CHAT#15 TEXT <text>

User exiting channel display. This text is shown to other users on a chat channel when another user exits the channel, bound for someplace other than another chat channel.

CHAT#16 TEXT <text>

Acknowledge public message – recipient available. This text is shown to a user when he enters a message in a public chat channel, and there are other users there who can see the message he typed in.

CHAT#17 TEXT <text>

Acknowledge public message – recipient not available. This text is shown to a user when he enters a message in a public chat channel, and there are no other users there to see the message that was typed. Ordinarily, this text is used to alert the the user that his chat message went to nobody.

CHAT#18 TEXT <text>

Acknowledge exit from private channel. This text is shown to a user when he exits a private chat with another user, to acknowledge that the exit took place.

CHAT#19 TEXT <text>

User hang-up alert. This text is shown to other users in a chat channel when a user hangs-up in chat. Usually used for “user vanished” displays and such.

CHAT#20 TEXT <text>

Private channel prompt. Related to CHAT#12_TEXT and CHAT#13_TEXT, this text is shown to a user when they are in a private chat channel as the input prompt. When this is not used, there is no prompt when in a private channel.

CHAT#21 TEXT <text>

Chat prompt used on high channel numbers. Related to CHAT#12_TEXT, CHAT#13_TEXT and CHAT#20_TEXT, this is the prompt shown to a user when they are in a chat channel numbered higher than the number set with the PUB_CHANNEL_NUM configuration statement.

CHAT#22 TEXT <text>

Chat prompt used on lowest channel. Related to CHAT#12_TEXT, CHAT#13_TEXT, CHAT#20_TEXT, and CHAT#21_TEXT, this is the prompt shown to a user when they are in a chat channel equal to the one set with the MIN_CHANNEL_NUM configuration statement.

CHAT#23 TEXT <text>

Preface for buffered mode chat message displays. This text is shown to preface incoming chat messages whenever chat messages are buffered. (Messages are buffered within channels, but not within private one-on-one chat.) Generally, this is simply a carriage return (^M).

CHAT#24 TEXT <text>

User you're waiting on logged off message. Applies only when a user enters chat in "wait mode" (either the /E:1 switch, or from a CHAT#01_CMD command). This text is shown to a user who is waiting on another user, and that other user logs off the system prior to coming to chat.

General Error Text Configuration Statements

ERROR#01 TEXT <text>

Bad command input error. This text is displayed when a user enters a command within the chat environment that UltraChat cannot decipher.

ERROR#02 TEXT <text>

User not online error. This text is displayed when a user executes a chat command which references a user who must be logged onto the system in order for the command to work, and the user referenced is not online.

ERROR#03 TEXT <text>

User name or ID ambiguous error. This text is displayed when a user executes a chat command which references a user name or ID, but the name or ID entered is not specific enough for UltraChat to reference it to a user. This applies in some cases, since UltraChat does not require a user to enter a complete name or ID, only a partial match. In essence, this error means that a partial match was made with more than one user.

ERROR#04 TEXT <text>

Who command executed while user invisible error. This text is displayed when a user attempts to execute a who's online display, but that user is marked as invisible.

ERROR#05 TEXT <text>

Chat message sent to user in no chat mode error. This text is shown to a user who sends a private message (PRIVATE#xx_CMD) to another user, and that other user is currently in no chat mode.

ERROR#06 TEXT <text>

Command on yourself error. This text is shown to a user who attempts to execute one of various commands which reference a specific user, and the user specified himself as the target user for the command.

ERROR#07 TEXT <text>

User name or ID ambiguous, but one exact match error. This text is displayed when a user executes a chat command which references a user name or ID, and the user name or ID given does match at least one user exactly but still may apply to more than one user. This applies in some cases, since UltraChat does not require a user to enter a complete name or ID, only a partial match. In essence, this error means that a partial match was made with more than one user. This text should tell the user to end the user name with a colon (:) to designate "exact match with this user." Refer to the sample configurations supplied with UltraChat for specific applications.

ERROR#08 TEXT <text>

Public message attempted when in read only mode error. This text is shown to a user who is in the public chat channel in read-only mode, and the user attempts to enter a public message (which is prohibited from read-only mode).

Miscellaneous Text Configuration Statements

The following statements allow you to define various text strings of a miscellaneous nature:

MISC#01 TEXT <text>

No queued messages available. This text is shown to a user who has executed UltraChat with a /QD switch on the Opt Data of the calling menu entry, and there are not queued messages to be displayed.

MISC#02 TEXT <text>

Press any key prompt. This text is shown by UltraChat in any area where the display is paused pending a user's keypress. In most cases, this simply says "Press any key to continue."

MISC#03 TEXT <text>

Logon announce text. This text is shown to other users who are in a window of opportunity alerting them that a user has logged onto the system, i.e., this is the logon announcement string.

MISC#04 TEXT <text>

Header string for read prompt entry. This text is shown to a user who types a forward slash (/) at a TBBS message read prompt to enter the chat environment.

MISC#05 TEXT <text>

Exit chat message. This text is shown when a user leaves the chat environment by any route other than the TBBS message read prompt.

MISC#06 TEXT <text>

Exit chat message (from read prompt). This text is shown when a user leaves the chat environment and return back to the TBBS message read prompt.

MISC#07 TEXT <text>

Removed from private channel message. This text is shown to other users when they are kicked out of a private channel, either with the "uninvite" command, or when the hosting user logs off or drops carrier.

MISC#08 TEXT <text>

Kill line acknowledgment. This text is shown to a user when they enter the keystroke designated as the “kill line” keystroke, defined with the `KILL_LINE_KEY` configuration statement.

MISC#09 TEXT <text>

Public chat banner. This text is shown to the user once when they enter a public chat channel.

MISC#10 TEXT <text>

Logon announcement generated alert. This text is shown to a user when their logon is announced to other users online.

MISC#11 TEXT <text>

Two minute warning. This text is shown to a user when their chat time limit is two minutes from expiring.

MISC#12 TEXT <text>

One minute warning. This text is shown to a user when their chat time limit is one minute from expiring.

MISC#13 TEXT <text>

Time expires warning. This text is shown to a user when their chat time limit has expired.

MISC#14 TEXT <text>

Two minute warning. This text is shown to a user when their chat inactivity timeout or chat time limit is two minutes from expiring.

MISC#15 TEXT <text>

One minute warning. This text is shown to a user when their chat inactivity timeout is one minute from expiring.

MISC#16 TEXT <text>

Time expired warning. This text is shown to a user when their chat inactivity timeout has expired.

Color Changing Statements

Come UltraChat displays are have color settings associated with them. These configuration statement allow you to set these colors by using ANSI escape sequences. (ANSI escape sequences are listed toward the end of Chapter 2 in your TBBS manual.)

IMPORTANT NOTE: You can override these color settings on a text display by text display basis by simply embedding ANSI escape sequences into any text string in the UltraChat configuration file.

COLOR#01 TEXT <ansi sequence>

Local typing color. This is the color of the typing performed by a user within the chat environment.

COLOR#02 TEXT <ansi sequence>

Remote typing color. This is the color of the typing performed by other users within the chat environment.

COLOR#03 TEXT <ansi sequence>

System typing color. This is the color of the typing performed by the system itself (error messages, who's online displays, etc.).

COLOR#04 TEXT <ansi sequence>

Highlight color. This is the color that will be used to highlight the typing of another user when the highlight command is executed.

Channel Changing Commands

UltraChat uses the name channels to express or designate where a user is in the chat environment at a given moment. Although often labeled "channels" to users, some chat systems do not refer to these places by that name. Sometimes they are conferences, chat sections, forums, etc. Whether you call them channels or not as far as your users are concerned, places to be in the chat environment are referred to and configured by that name within UltraChat.

All the channel changing commands follow this syntax:

CHANNEL#xx_CMD	Command
CHANNEL#xx_CONFIRM	Response
CHANNEL#xx_CHANPMT	Prompt
CHANNEL#xx_PWPMT	Prompt
CHANNEL#xx_ENTRY	Remote Display
CHANNEL#xx_EXIT	Remote Display
CHANNEL#xx_BADNUM	Error Message
CHANNEL#xx_NOTNUM	Error Message
CHANNEL#xx_NOTINV	Error Message
CHANNEL#xx_BADPW	Error Message

Where "xx" is the channel command number from 01 to 07 (see below for specifics).

CHANNEL#01_CMD

Channel change. CHANNEL#01_CMD is the basic channel change command, and allows a user to change from one public channel to another public channel.

CHANNEL#02_CMD

Private channel toggle. CHANNEL#02_CMD switches a user back and forth, in and out of their own private channel and the public channel they were last in.

CHANNEL#03_CMD

Goto another user's private chat channel. CHANNEL#03_CMD accepts a user's name and goes to that private chat channel, assuming that they are allowed (have been invited) to be there.

CHANNEL#04_CMD

Goto another user's private chat channel with password. CHANNEL#04_CMD is identical in function to CHANNEL#03_CMD, except that it accepts a password and performs a comparison against the password set for the private channel they want to go to. Assuming the password matches, and the user is otherwise allowed (have been invited), they are switched to that channel.

CHANNEL#05_CMD

Goto another user's private chat channel with invite override. CHANNEL#05_CMD operates just like CHANNEL#03_CMD, except that it ignores the invite status and goes to the desired channel without checking whether they are allowed (have been invited). Primarily of value to sysops.

CHANNEL#06_CMD

Goto another user's private chat channel with password and with invite override. CHANNEL#06_CMD operates just like CHANNEL#04_CMD, except that it ignores the invite status and goes to the desired channel without checking whether they are allowed (have been invited). Primarily of value to sysops.

CHANNEL#07_CMD

Go to the channel you were in when you entered chat. This command allows a user to return to the channel they were in initially when they entered chat. It is of particular interest to a user who entered chat in a named conference, then went to a numbered channel, and wants to return to the named conference. For example, this is similar to the /Q command in TBBS chat when that command is executed from a CB channel.

CHANNEL#xx CONFIRM

Command confirmation (response). This is the text shown to the user who is issuing the channel change command to confirm that the channel change command has been successfully executed. It is issued after the channel change command is completed. It applies to all CHANNEL#xx_CMD commands.

CHANNEL#xx CHANPMT

Channel entry prompt (prompt). This is the prompt shown to the user to ask for the channel to go to. It is used with all CHANNEL#xx_CMD commands. The user's input may be a channel number (as for the CHANNEL#01_CMD command), or a user name (as for the CHANNEL#03_CMD and similar commands).

CHANNEL#xx PWPMT

Password entry prompt (prompt). This is the prompt shown to the user to ask for a password before moving to the desired private channel. It is used with the CHANNEL#04_CMD command.

CHANNEL#xx ENTRY

Channel entry notification (remote display). This is the text shown to other users in the channel when the user issuing the channel change command enters the channel they switched to. It applies to all CHANNEL#xx_CMD commands.

CHANNEL#xx EXIT

Channel exit notification (remote display). This is the text shown to other users in the channel when the user issuing the channel change commands leaves the channel they are in to go to another channel. It applies to all CHANNEL#xx_CMD commands.

CHANNEL#xx BADNUM

Invalid number entered (error message). This error message is shown to the user when they enter an invalid channel number when prompted. It is used with the CHANNEL#01_CMD command.

CHANNEL#xx NOTNUM

Input was not numeric (error message). This error message is shown to the user when they enter a non-numeric value when prompted for a channel number. It is used with the CHANNEL#01_CMD command.

CHANNEL#xx NOTINV

Not invited error (error message). This error message is shown to the user when they attempt to go a private chat channel to which they have not been invited. It applies to the CHANNEL#03_CMD and CHANNEL#04_CMD commands.

CHANNEL#xx BADPW

Bad password error (error message). This error message is shown to the user when they attempt to go to a private, password-protected chat channel, but the password they provide does not match the password set for the desired private channel. It applies to the CHANNEL#04_CMD command.

General Chat Commands

While within the chat environment, UltraChat offers a selection of commands to take a user into chat mode from other operating modes of UltraChat, for the general purpose of going to one-on-one private chat. These general chat commands allow you to tailor chat entry to your own needs. All the general chat commands follow this syntax:

CHAT#xx_CMD	Command
CHAT#xx_CONFIRM	Response
CHAT#xx_PROMPT	Prompt
CHAT#xx_EXIT	Remote Display
CHAT#xx_REMOTE	Remote Display
CHAT#xx_NOCHAT	Error Message
CHAT#xx_SELF	Error Message

Where "xx" is the general chat command number from 01 to 06 (see below for specifics).

CHAT#01 CMD

Ask a user to chat, and wait for that user. This command is designed to emulate the functionality of the Oracomm /BRING command within the chat environment. It allows a user to issue a command to request private chat of another user. The target user is alerted that someone wants to chat with them. The user who makes the request goes into a waiting mode for the target user to join them in chat. If the target user chooses to respond to the chat request, both users will be placed in one on one private chat mode, and will remain there until they exit.

CHAT#02 CMD

Ask a user to chat, and proceed into the chat environment. This command is designed to emulate the functionality of the Oracomm /CALL command within the chat environment. It allows a user to issue a command to request chat of another user. The target user is alerted that someone wants to chat with them. The user who makes the request can proceed to the usual chat facilities (enter public chat, etc.) and does not explicitly wait for the target user to join them.

CHAT#03 CMD

Enter main public chat channel, Oracomm style. This command is designed primarily to allow entry to the chat environment from the TBBS message read prompt. It enters chat in an Oracomm-style mode, where the user is shown a list of who is online, and can either go to public chat or select a user to chat with by number from the list shown. (The user could also abort the chat entry if desired.)

CHAT#04 CMD

Enter main public chat channel, directly. This command is designed primarily to allow entry to the chat environment from the TBBS message read prompt. It enters the chat environment directly, taking the user to the main public chat channel without any interim steps.

CHAT#05_CMD

Ask a user to chat, and wait for that user while still operating. This command is identical in function to the CHAT#01_CMD command, but instead of forcing the user to wait for the other user to come chat, it allows the requesting user to continue executing chat commands. The requesting user is yanked immediately into chat when the remote user accepts the chat request. If the requesting user leaves the chat environment, the command is terminated.

CHAT#06_CMD

Ask a user to chat, and wait for that user while still operating, 2 minute limit. This command is identical in function to the CHAT#05_CMD command, except that the requesting user will wait only 2 minutes for the other user to accept, at which time they are sent a message saying that the other user is not joining them and they are no longer waiting. (The text of this message is contained in the MSC#18_TEXT configuration statement.)

CHAT#xx_CONFIRM

Message sent confirmation (response). This response applies only to the CHAT#01_CMD command. This text is shown to the user to confirm that his CHAT#xx_CMD command was successful, and that the message given was sent to the specified user in conjunction with his command.

CHAT#xx_PROMPT

Message input prompt (prompt). This prompt applies only to the CHAT#01_CMD and CHAT#02_CMD commands. It is shown to the user to request that the user enter the desired message to send in conjunction with the command. Once entered, the message is sent to the user who is the target of the command.

CHAT#xx_EXIT

User left chat announcement (remote display). This remote display applies only to the CHAT#01_CMD and CHAT#02_CMD commands. It is shown when the user issuing the corresponding com-

mand exits the chat channel, either entirely, or when moving to another channel.

CHAT#xx REMOTE

Chat request message (remote display). This remote display applies only to the CHAT#01_CMD command. It is shown to a target user when another user executes a CHAT#01_CMD command, and alerts that target user that the initiating user wants to chat with them.

CHAT#xx NOCHAT

Target user is in no chat mode (error message). This error message applies only to the CHAT#01_CMD command. It is displayed to the initiating user when they issue a CHAT#01_CMD command, and the target user is currently in “no chat” mode and unable to accept the request.

CHAT#xx SELF

Target user is the same as initiating user (error message). This error message applies only to the CHAT#01_CMD command. It is displayed to the initiating user when they issue a CHAT#01_CMD command, and they are the target user, i.e., the user is trying to chat with himself.

Private Message Commands

UltraChat offers a variety of ways to send private messages – messages sent by one user to another specific user that nobody but the sender and recipient can see. Private messages are used for user to user paging, so-called whisper commands, and so on. UltraChat offers nine (9) functionally identical private message commands, which allows you to configure to up nine (9) different styles or types of private messaging in the chat environment.

All the private message commands follow this syntax:

PRIVATE#xx_CMD	Command
PRIVATE#xx_CONFIRM	Response
PRIVATE#xx_NAMEPMT	Prompt
PRIVATE#xx_MSGPMT	Prompt
PRIVATE#xx_EMPTY	Remote Display
PRIVATE#xx_REMOTE	Remote Display
PRIVATE#xx_NOCHAT	Error Message
PRIVATE#xx_ERROR	Error Message

Where “xx” is the private command number from 01 to 10 (see below for specifics).

PRIVATE#01_CMD – PRIVATE#09_CMD

Private message sending. The PRIVATE#01_CMD through PRIVATE#09_CMD commands are functionally identical, and all are interchangeable, allowing you to configure up to nine (9) separate private message sending modes or types. All allow a user to send a private (not displayed in public chat) message to another user online. If the receiving user is in the chat environment, they receive the message immediately. If the receiving user is outside of the chat environment, they will receive the message at the next window of opportunity.

PRIVATE#10_CMD

Private message resend. This command is identical to the PRIVATE#01_CMD through PRIVATE#09_CMD commands – except that no user name or ID is required to be entered by the user. Instead, it sends a private message to the last person the user send a private message to. It is intended to allow you to provide a shortcut command that gives users an easy, quick way to converse repeatedly with another user privately while still participating normally in a public chat channel.

PRIVATE#xx_CONFIRM

Command confirmation (response). This is the text shown to the user who is issuing the private message command to confirm that

the private message command has been successfully executed. It is issued after the private message is delivered to the receiving user.

PRIVATE#xx NAMEPMT

User name prompt (prompt). This is the prompt shown to the user to ask for the name (or other identifier) of the user they wish to send the private message to, if the name (or identifier) was not provided as part of the command.

PRIVATE#xx MSGPMT

Message content prompt (prompt). This is the prompt shown to the user to ask for the contents of the private message they wish to send, if the message content was not provided as part of the command.

PRIVATE#xx EMPTY

No message content provided display (remote display). When present, this command element allows a user to send a private message that has no message content. When the user sends a private message with no content, the text associated with this command element is sent instead. If this command element is not defined, UltraChat requires the user to type message content in order for a private message to be sent.

PRIVATE#xx REMOTE

Remote user display (remote display). This is the text shown to the remote user when they receive a private message from someone.

PRIVATE#xx NOCHAT

Receiving user in no chat mode error (error message). This is the text shown to the user issuing the private message command when the user they are sending the message to (the receiving user) is currently in no chat (ignore) mode.

PRIVATE#xx ERROR

Remote user not available error (error message). This is the text shown to the user issuing the private message command when the user they are sending the message to (the receiving user) is not available (not logged in, etc.).

Who's Online Commands

UltraChat offers a variety of ways for you to provide who's online and related displays. These displays all list various information about other users online or in the TBBS userlog. Used primarily for true "who" commands, these commands also provide a variety of user listing functions. All the who's online commands follow this syntax:

WHO#xx_CMD	Command
WHO#xx_HEADER	Response
WHO#xx_TOTAL	Response
WHO#xx_NORMAL	Response
WHO#xx_IDLE	Response
WHO#xx_INVISIBLE	Response
WHO#xx_PAUSE	Response
WHO#xx_PUBLIC	Response
WHO#xx_MYCHAN	Response
WHO#xx_HISCHAN	Response
WHO#xx_PRIVATE	Response
WHO#xx_NOUSERS	Response
WHO#xx_NAMEPMT	Prompt
WHO#xx_ERROR	Error Message
WHO#xx_ANON	Error Message

Where "xx" is the who's online command number from 01 to 13 (see below for specifics).

Most who's online displays use repeated, sequential responses to form columnar displays. For example, the WHO#01_CMD command will display a response (usually the WHO#01_NORMAL text) once for every line on the system – up to the line number designated by the LAST_LINE_NUM configuration statement. This is unlike other types of UltraChat commands that generally

display applicable response strings once each time the command is executed.

WHO#01_CMD -- WHO#06_CMD

General who's online command. The WHO#01_CMD through WHO#06_CMD commands are functionally identical, and are designed to allow you to provide up to six (6) different styles of who's online user listing displays. All of these who's online commands will reference all lines on your system – up to the line number designated by the LAST_LINE_NUM configuration statement. You can control which types of lines are displayed (in use, idle, etc.) through command element definitions.

WHO#07_CMD

Current channel who's online listing. This command lists the users who are on the same chat channel as the user issuing the who's online command.

WHO#08_CMD

Who's online and using chat listing. This command lists all users who are currently logged onto the system and who are within the chat environment, i.e., those users who are using chat (whether they are in the current channel or not).

WHO#09_CMD

Browse the user who sent this message. This command applies only when executed from a TBBS message read prompt. It performs a browse command (normally made equivalent to the WHO#10_CMD command) on the user who sent the message the user is currently reading.

WHO#10_CMD

Browse a user. This command is designed to allow a user to browse another user on the system. It is normally used to display the three-line bio of another user.

WHO#11_CMD

List all users in reverse alphabetical order. This command is designed to allow a user to list all users on the system in reverse alphabetical order.

WHO#12_CMD

List all users in forward alphabetical order. This command is designed to allow a user to list all users on the system in forward alphabetical order.

WHO#13_CMD

List new system users. This command is designed to allow a user to list all new users on the system. The `NEW_USERS_NUM` configuration command tells UltraChat how to control the scope of this listing.

WHO#xx_HEADER

Header for who listing (response). This response applies to the `WHO#01_CMD` through `WHO#06_CMD`, and `WHO#08_CMD` commands only. This text is the first thing shown to a user when the corresponding who's online command is executed. If the display is arranged in a columnar fashion (as it usually is), then this text should feature headings for the various columns.

WHO#xx_TOTAL

Totals line for who listing (response). This response applies to the `WHO#01_CMD` through `WHO#08_CMD` commands only. This text is the last thing shown to a user when the corresponding who's online command is executed. Generally it is used to provide a summary or totals, such as how many people are online at the moment ("12 of 24 lines in use") and the like.

WHO#xx_NORMAL

Listing display for normal user (response). This response applies to all `WHO#xx_CMD` commands. This text forms the basis for all who's online displays. It contains whatever text is shown to the user

who executed the who's online command, i.e., the actual contents of the who's online display. It is shown for every line that has a user that is not invisible.

WHO#xx IDLE

Listing display for idle user (response). This response applies to the WHO#01_CMD through WHO#06_CMD commands only. This text is in place of the WHO#xx_NORMAL response for any lines that are idle (no user logged on). If left blank, nothing is displayed for idle lines.

WHO#xx INVISIBLE

Listing display invisible user (response). This response applies to the WHO#01_CMD through WHO#06_CMD commands only. This text is in place of the WHO#xx_NORMAL response for any line that have an invisible user logged on. If left blank, nothing is displayed for lines with invisible users.

WHO#xx PAUSE

"P to pause, S to stop" display (response). This response applies to the WHO#11_CMD through WHO#13_CMD commands only. For the applicable who's online commands, this text is shown prior to any other text, and is designed to allow the system to instruct the user to press P to pause the display, or S to stop the display, since the displays associated with these commands can run on screen after screen. NOTE: The P and S keys can be changed using configuration statements if desired.

WHO#xx PUBLIC

Header when in public channel (response). This response applies only to the WHO#07_CMD command. When the corresponding command is used from within a public chat channel or named conference, this text is the first thing shown to the user in conjunction with the command. It forms a header for the display information that follows it. If the display is arranged in a columnar fashion (as it usually is), then this text should feature headings for the various columns.

WHO#xx MYCHAN

Header when in own private channel (response). This response applies only to the WHO#07_CMD command. When the corresponding command is used from within the user's own private chat channel, this text is the first thing shown to the user in conjunction with the command, and replaces the WHO#xx_PUBLIC response normally displayed. It forms a header for the display information that follows it. If the display is arranged in a columnar fashion (as it usually is), then this text should feature headings for the various columns.

WHO#xx HISCHAN

Header when in someone else's private channel (response). This response applies only to the WHO#07_CMD command. When the corresponding command is used from within someone else's private chat channel, this text is the first thing shown to the user in conjunction with the command, and replaces the WHO#xx_PUBLIC response normally displayed. It forms a header for the display information that follows it. If the display is arranged in a columnar fashion (as it usually is), then this text should feature headings for the various columns.

WHO#xx PRIVATE

Header when in private one-on-one chat (response). This response applies only to the WHO#07_CMD command. When the corresponding command is used from within private one-on-one chat, this text is the first thing shown to the user in conjunction with the command, and replaces the WHO#xx_PUBLIC response normally displayed. It forms a header for the display information that follows it. If the display is arranged in a columnar fashion (as it usually is), then this text should feature headings for the various columns.

WHO#xx NOUSERS

Totals line for who listing when no users online (response). This response applies only to the WHO#07_CMD command. This text is the last thing shown to a user when the corresponding who's online command is executed, and is used in replacement of the WHO#xx_TOTAL line when there are no users to total.

WHO#xx NAMEPMT

User name prompt (prompt). This prompt applies only to the WHO#10_CMD command. It is given to the user when they issue the corresponding who's online command and neglect to provide a user name or ID with the command. It asks the user to enter the name or ID of the user they wish to browse.

WHO#xx ERROR

User not found error (error message). This error message applies only to the WHO#09_CMD and WHO#10_CMD commands. It is given when the user designates a user name or ID that UltraChat cannot find in conjunction with the who's online command.

WHO#xx ANON

Command executed on anonymous message board error (error message). This error message applies only to the WHO#10_CMD command. It is given when the user wants to browse another user from the message read prompt, but the browse command is executed from an anonymous message board (where user information is confidential).

Chat Exit Commands

UltraChat offers a variety of ways to leave the chat environment. The commands that follow all perform this function in different ways. All the chat exit commands follow this syntax:

EXIT#xx_CMD Command

None of the chat exit commands have any command elements.

EXIT#01_CMD

Exit to BBS. This command allows a user to completely exit the chat environment and return to TRBS control. This is the basic, standard "exit chat" command.

EXIT#02 CMD

Exit private chat. This command puts the user "up" a level within the chat environment (and does not perform an exit from the chat environment). When a user is in private chat with someone, they will go back to the last public chat channel they were in before entering private chat. If the user is already in public chat, the command is ignored.

EXIT#03 CMD

Exit via logoff. This command performs the equivalent of a TBBS Type 10 command, logging the user off with the usual logoff status displays. With this command, the user leaves not only the chat environment, but the BBS as well.

EXIT#04 CMD

Exit via carrier drop. This command immediately drops carrier on the user, causing them to disappear from the system. Please note that TBBS will still send logoff status displays to the modem before resetting it, so you should keep logoff information as short as possible so that the line will become available again as quickly as possible. With this command, the user leaves not only the chat environment, but the BBS as well.

Help File Display Commands

UltraChat gives you the flexibility to designate up to nine (9) help files. Users can issue commands to display the help files you configure. You may, for example, wish to have one help file for general chat commands usage, and others for using specific features (such as private chat, the three-line bio, etc.). All the help file display commands follow this syntax:

HELP#xx_CMD	Command
HELP#xx_FILE	Response
HELP#xx_ERROR	Error Message

Where "xx" is the help file display command number from 01 to 09 (see below for specifics).

HELP#01_CMD – HELP#09_CMD

Help file display. The **HELP#01_CMD** through **HELP#09_CMD** commands are functionally identical, and all are interchangeable, allowing you to configure up to nine (9) separate help file displays. All allow a user to display a help file online upon issuing the corresponding help keystroke. During a help file display, all UltraChat messages will be suppressed.

HELP#xx_FILE <filename>

Help file name (response). This command element allows you to designate the drive, path and filename of the help file associated with the help command. When the user issues the corresponding help command, this file will be opened and displayed. During the help file display, all messages from UltraChat are suppressed.

HELP#xx_ERROR

File missing error (error message). This element allows you to designate text that is shown to the user when the help file is missing (or cannot be located by UltraChat). In any situation where the user attempts to display a help file, and the help file cannot be located, this error message text will be displayed in lieu of the file.

Text Display Commands

UltraChat allows you to configure up to nine (9) text strings which can be associated to a chat command and display every time that command is issued. These are usually used for information displays, such as chat version, system name, etc. By using insertion parameters, you can also use them to have commands that display the current time and date, or other dynamic information. All the text display commands have this syntax:

TYPE#xx_CMD	Command
TYPE#xx_TEXT	Response

Where "xx" is the text display command number from 01 to 09 (see below for specifics).

TYPE#01_CMD - TYPE#19_CMD

Generic text display commands. The TYPE#01_CMD through TYPE#19_CMD commands are functionally identical, and all are interchangeable, allowing you to configure up to nineteen (19) separate text displays. All allow a user to display a text string online upon issuing the corresponding help keystroke.

TYPE#xx TEXT

Text string to display (response). This command element lets you designate the text string that will be displayed to the user when the corresponding command is issued by a user. This is standard text, and can include insertion parameters for dynamic displays.

Status Toggle (On to Off, Off to On) Commands

UltraChat offers a variety of toggles, which allow the user to toggle the state of a status item within the chat environment. Toggling means that if something is turned off (disabled), it will be turned on (enabled). If it's turned on (enabled), it will be turned off (disabled). In other words, the current state of the status item will be changed or toggled.

The status items that can be toggled include: a user's invisible status, a user's no chat (ignore) status, logon announcement receipt status, shortcut commands allowed status, and moan macro status. All the status toggle commands follow this syntax:

TOGGLE#xx_CMD	Command
TOGGLE#xx_ON	Response
TOGGLE#xx_OFF	Response

Where "xx" is the status toggle command number from 01 to 05 (see below for specifics).

TOGGLE#01_CMD

Toggle invisible status. This command allows a user to toggle his invisible status. When invisible, a user will not show up in any who's online listings under the control of UltraChat until they toggle the

status off. The invisible state of a user is saved between calls. Note that a user cannot issue a who's online command while invisible, and will be made visible upon entry to chat.

TOGGLE#02_CMD

Toggle no chat status. This command allows a user to toggle his no chat status. When no chat status is on (enabled), the user will show on who's online listings, but will not be available to other users in the chat environment for paging, etc. until they toggle the status off.

TOGGLE#03_CMD

Toggle logon announcement suppression. This command allows a user to toggle the logon announcement suppression status. When this status is enabled, logon announcements generated by Ultra-Chat will be suppressed. (Logon announcements are messages broadcast to users to notify them that someone has logged onto the system.) Logon announcements remain suppressed until the status is toggled off.

TOGGLE#04_CMD

Toggle shortcut commands required. This command allows a user to toggle whether shortcut chat commands are required or not. When this status is enabled, the user must use shortcut commands in the chat environment. When this status is disabled, long form commands as well as shortcut commands are allowed within the chat environment. This is useful if a user wants to make sure what they type will not be interpreted as a command.

TOGGLE#05_CMD

Toggle moan macro status. This command allows a user to toggle whether they can use moan macro commands in the chat environment. When enabled, the user can use any defined moan macros they have access to. When this status is disabled, moan macros cannot be issued by the user.

TOGGLE#xx ON

Status enabled confirmation (response). This is the text shown to the user when a status item is toggled on (enabled) to confirm that the toggle command was successful and to indicate the new state of the corresponding status item.

TOGGLE#xx OFF

Status disabled confirmation (response). This is the text shown to the user when a status item is toggled off (disabled) to confirm that the toggle command was successful and to indicate the new state of the corresponding status item.

Status Forced On (Enable) Commands

Besides the toggles described previously, UltraChat offers equivalents for all the toggles that force a given status item on (enabled). The status item is forced on, regardless of its current state. If it's currently off, it will be turned on; if it's already on, forcing it on will result in no change.

The status items that can be forced on include: a user's invisible status, a user's no chat (ignore) status, logon announcement receipt status, shortcut commands allowed status, and moan macro status. All the status forced on commands follow this syntax:

ON#xx_CMD	Command
ON#xx_TEXT	Response

Where "xx" is the status forced on command number from 01 to 05 (see below for specifics).

ON#01 CMD

Force on invisible status. This command allows a user to force on his invisible status. When invisible, a user will not show up in any who's online listings under the control of UltraChat until they turn the status off. The invisible state of a user is saved between calls. Note that a user cannot issue a who's online command while invisible, and will be made visible upon entry to chat.

ON#02 CMD

Force no chat status. This command allows a user to force on his no chat status. When no chat status is on (enabled), the user will show on who's online listings, but will not be available to other users in the chat environment for paging, etc. until they turn the status back off.

ON#03 CMD

Force suppress logon announcements. This command allows a user to force on the logon announcement receive status. When this status is enabled, logon announcements generated by UltraChat are suppressed. (Logon announcements are messages broadcast to users to notify them that someone has logged onto the system.)

ON#04 CMD

Force on shortcut commands required. This command allows a user to force on the shortcut chat commands required status. When this status is on (enabled), the user must use shortcut commands in the chat environment. When this status is off (disabled), long form commands as well as shortcut commands are allowed within the chat environment.

ON#05 CMD

Force on moan macro status. This command allows a user to force on whether they can use moan macro commands in the chat environment. When on (enabled), the user can use any defined moan macros they are allowed to use. When this status is off (disabled), moan macros cannot be issued by the user.

ON#xx TEXT

Status enabled confirmation (response). This is the text shown to the user when a status item is forced on (enabled) to confirm that the command was successful and to indicate the new state of the corresponding status item.

Status Forced Off (Disable) Commands

Besides the toggles described previously, UltraChat offers equivalents for all the toggles that force a given status item off (disabled). The status item is forced off, regardless of its current state. If it's currently on, it will be turned off; if it's already off, forcing it off will result in no change.

The status items that can be forced off include: a user's invisible status, a user's no chat (ignore) status, logon announcement receipt status, shortcut commands allowed status, and moan macro status. All the status forced on commands follow this syntax:

```
OFF#xx_CMD      Command  
OFF#xx_TEXT Response
```

Where "xx" is the status forced off command number from 01 to 05 (see below for specifics).

OFF#01 CMD

Force off invisible status. This command allows a user to force off his invisible status, making the user visible. When visible, a user will show up in any who's online listings under the control of UltraChat until they turn the status on again (at which point they become invisible and not listed in who's online displays). The invisible state of a user is saved between calls.

OFF#02 CMD

Force off no chat status. This command allows a user to force off his no chat status. When no chat status is off (disabled), the user will show on who's online listings and will be available to other users in the chat environment for paging, etc. until they turn the status back on. (When the status is on, the user is not available to other users.)

OFF#03 CMD

Force off logon announcement suppression (allow announcements). This command allows a user to force off the logon announcement receive status. When this status is disabled, logon

announcements generated by UltraChat are shown to the user. (Logon announcements are messages broadcast to users to notify them that someone has logged onto the system.)

OFF#04 CMD

Force off shortcut commands required. This command allows a user to force off the requirement to use shortcut chat commands in the chat environment. When this status is on (enabled), the user must use shortcut commands in the chat environment. When this status is off (disabled), long form commands as well as shortcut commands are allowed within the chat environment.

OFF#05 CMD

Force off moan macro status. This command allows a user to force off whether they can use moan macro commands in the chat environment. When on (enabled), the user can use any defined moan macros they are allowed to use. When this status is off (disabled), moan macros cannot be issued by the user.

OFF#xx TEXT

Status disabled confirmation (response). This is the text shown to the user when a status item is forced off (disabled) to confirm that the command was successful and to indicate the new state of the corresponding status item.

Three-Line Bio Commands

UltraChat offers support for the entry and storage of a three-line bio on a user by user basis. This feature, patterned after similar features in other BBS chat systems, allows you to conveniently and easily offer brief user profiles to your callers. The three-line bios can then be displayed as a part of "who's online" displays, or can be looked-up by users online from within the chat environment. UltraChat offers two commands to allow the entry and manipula-

tion of the three-line bio feature. The three-line bio commands both follow this syntax

BIO#xx_CMD	Command
BIO#xx_YN	Prompt
BIO#xx_PROMPT	Prompt

Where “xx” is the three-line bio command number from 01 to 02 (see below for specifics).

BIO#01_CMD

Set initial three-line bio. This command allows a user to set their three-line bio for the first time. The prompts (described below) should reflect that this command is used when the user is setting their bio for the first time.

BIO#02_CMD

Change three-line bio. This command allows a user to change their existing three-line bio. This command should be available to users conveniently if you use the three-line bio feature on your system. The prompts (described below) should reflect that this command is used for changing a bio, as opposed to setting it initially.

BIO#xx_YN

Ask yes/no whether to set/update (prompt). This prompt is shown to the user when they initiate one of the BIO#xx_CMD commands. It is designed to prompt the user whether to proceed with the command. It expects a yes/no response from the user. If they answer yes, then the command continues to be executed. If they answer no, the command is aborted.

BIO#xx_PROMPT

Ask for three-line bio (prompt). This prompt is shown to the user when they initiate one of the BIO#xx_CMD commands, and have confirmed (through the BIO#xx_YN prompt) that they wish to proceed. This prompt should ask the user to enter their three-line bio. It should also give introductory information about the bio itself if applicable, and should note that the user should NOT press Enter

until they are done entering the bio. Refer to the supplied sample configurations for examples.

Edit Commands

UltraChat offers a variety of settable, editable options for users. These include announce strings (generally shown with who's online displays), private channel topic name, and private channel password. When you wish any of these features to be active and available for users, configure the desired command within UltraChat. All the edit commands have the following syntax:

EDIT#xx_CMD	Command
EDIT#xx_PROMPT	Prompt
EDIT#xx_CONFIRM	Response
EDIT#xx_ERROR	Error Message

Where "xx" is the edit command number from 01 to 04 (see below for details).

EDIT#01_CMD

Set or change announce string. This command allows a user to set or change their announce string. The announce string is generally used as a component of a who's online display, and is employed by users to customize what is shown by their name during a who's online display, and so on, acting as a user-settable vanity item.

EDIT#02_CMD

Set topic name for private chat channel. This command allows a user to specify a name or topic for their private chat channel. This name can then be used on who's online displays and the like to identify the topic, subject, name, etc. of a user's private channel. This is primarily a vanity item, but is often used by a caller to use their private channel to setup a particular chat topic and foster communication on that topic.

EDIT#03_CMD

Set password for private chat channel. This command allows a user to specify a password for their own private chat channel. When set, another user who wishes to go to that private channel must know and provide the password before being allowed access.

EDIT#04_CMD

Change user handle (name/ID override). This command allows a user to enter a handle that will be used within the chat environment, and which overrides the user's regular name or user ID for the duration of their session. Handles are saved between sessions so that a user can define a handle and keep it for on-going use. Handles are used exactly as entered by the user, and are not case-sensitive.

EDIT#xx_PROMPT

Prompt user for desired command input (prompt). This prompt asks the user to enter the information for the corresponding command. For example, with the **EDIT#03_CMD** command, this should prompt the user to enter their desired private chat channel password.

EDIT#xx_CONFIRM

Confirm acceptance of command input (response). This response is shown to the user when they have successfully entered the information expected by the corresponding command. For example, with the **EDIT#03_CMD** command, this should confirm that the password they input has been accepted.

EDIT#xx_ERROR

Handle entered is in use error (error message). This error message is shown when a user executes an **EDIT#04_CMD** command to enter or change their handle, and the handle they entered is the name or ID of a user on the system, or is the same as the handle of someone else who is online at the moment.

Interaction Commands

UltraChat offers a variety of interaction commands. These commands allow a user to control various aspects of their interaction with other users in the chat environment. Examples of this interaction include a user being able to ignore or squelch another user's chatter, invite or uninvite users to their private channel, and so on. All interaction commands follow this syntax:

INTERACT#xx_CMD	Command
INTERACT#xx_ON	Response
INTERACT#xx_ON2	Response
INTERACT#xx_OFF	Response
INTERACT#xx_OFF2	Response
INTERACT#xx_ALLON	Response
INTERACT#xx_ALLOFF	Response
INTERACT#xx_NAMEPMT	Prompt
INTERACT#xx_REMOVE	Remote Display
INTERACT#xx_REMOTEINUC	Remote Display
INTERACT#xx_REMOTE	Remote Display
INTERACT#xx_ERROR	Error Message

Where "xx" is the interaction command number from 01 to 07 (see below for details).

INTERACT#01_CMD

Ignore a user or all users. This command allows a user to mark another particular user, or all other users, as "ignored." If a user provides another user's name or ID when issuing this command, that one user is affected; without a name or ID given, this command will apply to all users. When a user is being ignored, none of their message traffic in the chat environment will be displayed, i.e., their chat will be suppressed. When ignored, a user's private chat commands as well as comments in a public channel will not be displayed to the user who initiated the ignore command. (The user's chat traffic is still shown to all others who are not ignoring that user.)

INTERACT#02_CMD

Remember a user or all users. This command is the functional opposite of the INTERACT#01_CMD command. It allows the

user to mark another particular user, or all users, as “remembered.” If a user provides another user’s name or ID when issuing this command, that one user is affected; without a name or ID given, this command will apply to all users. When a user is remembered (i.e., not ignored), all of their message traffic in the chat environment will be displayed (not suppressed).

INTERACT#03_CMD

Turn ignore status on and off. This command is closely related to the INTERACT#01_CMD and INTERACT#02_CMD commands. But where those commands force ignore or remember mode, this command allows the user to input which mode they desire. When issuing this command, a user may optionally provide a user name or ID, in which case the command applies only to that user. When no name or ID is given, the command applies to all users. Besides a user name or ID, the user can give the keyword “ON” or “OFF” that designates how the command is to behave, ON meaning to ignore, OFF meaning to remember. If no ON or OFF keyword is provided, ON (ignore) is assumed. Refer to the previous INTERACT#xx_CMD commands for an explanation of ignore and remember.

INTERACT#04_CMD

Invite a user or users to your private channel. This command allows a user to invite another user (or all users) to their private chat channel. (The UltraChat default is uninvited; this command allows a user to invite someone who previously was uninvited.) If a user follows this command with a user name or ID, the invite command applies only to that user. If no name or ID was specified, then all users are invited.

INTERACT#05_CMD

Uninvite a user or users from your private channel. This command is the functional opposite of the INTERACT#04_CMD command. Instead of inviting a user to come to your channel, it marks another user or users as uninvited, and thereby unable to come to your private channel. If a user is already in the private channel and they are uninvited, they are kicked out of the private channel and returned to the previous public channel they were in. If a user

issuing this command follows it with a user name or ID, the `uninvite` applies only to the named user. If they omit the name or ID, the `uninvite` command applies to all users.

INTERACT#06 CMD

Highlight a user's chatter. This command allows a user to "highlight" the chatter (chat traffic) of another user. This command expects the person issuing the command to provide the name or ID of the user whose chat traffic they wish to highlight. When a user is marked as highlighted, the color designated by the `COLOR#04_TEXT` is sent just before chat traffic from this user. `COLOR#04_TEXT` should, therefore, be configured to show text in a different color or intensity than normally used. A maximum of ten (10) users can be highlighted at one time.

INTERACT#07 CMD

Unhighlight a user's chatter. This command is the functional opposite of the `INTERACT#06_CMD` command. It allows a user to tell UltraChat to stop highlighting the chat traffic from the designated user. Like the `INTERACT#06_CMD` command, this command accepts the name or ID of a user whose chat traffic should stop being highlighted. If that user's traffic not highlighted already, this command will have no effect.

INTERACT#xx ON

User invitation/ignore response (response). This response is displayed to a user when they have invited another user to their private channel. This is the response shown when the invitation is made from the inviting user's private channel. Also used to acknowledge that a user has chosen to ignore another user.

INTERACT#xx ON2

User invitation not made from private channel response (response). This response applies only to the `INTERACT#04_CMD` and `INTERACT#05_CMD` commands. This text is shown to a user when he invites another user to his private channel, but that invitation is made when the inviting user is not currently located in his own private channel (i.e., made from public

channel or someone else's private channel). Also used to acknowledge that a user has chosen to ignore another user.

INTERACT#xx_OFF

User uninvite response (response). This response is displayed to a user when they have uninvited another user from their private channel. This is the response shown when the uninvite is made from the uninviting user's private channel. Also used to acknowledge that a user has chosen to remember another user.

INTERACT#xx_OFF2

User uninvite not made from private channel response (response). This response applies only to the INTERACT#04_CMD and INTERACT#05_CMD commands. This text is shown to a user when he uninvites another user from his private channel, but that uninvite is made when the uninviting user is not currently located in his own private channel (i.e., made from public channel or someone else's private channel). Also used to acknowledge that a user has chosen to remember another user.

INTERACT#xx_ALLON

All users invited response (response). This text is shown to a user when he invites all users to his private channel with a single command (normally an invite command that has no specific user designated). Also used to acknowledge that all users have been ignored.

INTERACT#xx_ALLOFF

All users uninvited response (response). This text is shown to a user when he uninvites all users from his private channel with a single command (normally an uninvite that has no specific user designated). Also used to acknowledge that all users have been remembered.

INTERACT#xx_NAMEPMT

User name prompt (response). This prompt is given to the user when they issue an invite or uninvite command and neglect to

provide a user name or ID with the command. It asks the user to enter the name or ID of the user they wish to invite or uninvite. NOTE: UltraChat will also accept the keyword "ALL" in lieu of a user name or ID, indicating to invite or uninvite all users, and this prompt should reflect that option if desired.

INTERACT#xx_REMOVE

User was uninvited and kicked out of channel (remote display). This remote display applies only to the INTERACT#04_CMD and INTERACT#05_CMD commands. This text is shown to a user who is in another user's private channel, and has just been uninvited from and removed from that channel by the other user (the user whose private channel it is). This text is ONLY shown to a user who has been uninvited when that user is already in the private channel he has been uninvited from. If the user is elsewhere (in a public channel, etc.) and gets uninvited, nothing is shown.

INTERACT#xx_REMOTEINUC

User invitation, in UltraChat (remote display). This remote display applies only to the INTERACT#04_CMD and INTERACT#05_CMD commands. This text is shown to a user who has been invited to another user's private channel. This particular text is shown if the user who is invited is within the chat environment (i.e., in UltraChat) already.

INTERACT#xx_REMOTE

User invitation not in UltraChat (remote display). This remote display applies only to the INTERACT#04_CMD and INTERACT#05_CMD commands. This text is shown to a user who has been invited to another user's private channel. This particular text is shown if the user who is invited is NOT within the chat environment (i.e., is not in UltraChat) already.

INTERACT#xx_ERROR

Invitation error (error message). This text is shown to a user when an error occurred during the handling of their invite/uninvite command. Such an error usually results from the entry of a name or ID

of a user who is not online when the invite or uninvite command is issued.

Action Commands

UltraChat supports action commands. This term refers to a set of special commands within the chat environment, which are a form of generic moan macro. In Major BBS chat, this is usually called a "generic action" or "GA" command. In essence, an action command allows a user to create on-the-fly a special, tailored remote display that will be shown to all the users who are in the channel with the user who issued the action command. In most ways an action command is the same as a user simply typing something into the chat channel – but the remote display is different, normally formatted as though the user issued a moan macro.

For example, in Major BBS mode, a user named Joe Blow might type the following:

ga is blushing in a pewky scarlet color...

and other users on the channel might see something like this:

Joe Blow is blushing in a pewky scarlet color...

Since you configure action commands, they can have whatever behavior you want them to have. They follow this syntax:

<code>ACTION#xx_CMD</code>	Command
<code>ACTION#xx_CONFIRM</code>	Response
<code>ACTION#xx_REMOTE</code>	Remote Display

Where "xx" is the action command number from 01 to 09 (see below for details).

ACTION#01_CMD – ACTION#09_CMD

Action commands. The ACTION#01_CMD through ACTION#09_CMD commands are all functionally identical. They allow you to designate up to nine (9) generic action commands

users can invoke to display customized text to other users in the chat channel.

ACTION#xx CONFIRM

Action command confirmation (response). This response is used to provide a confirmation to the user issuing the action command that the command was accepted and processed by UltraChat.

ACTION#xx REMOTE

Action command text shown to others (remote display). This is the text shown to other users in the chat channel of the user issuing the action command.

Executing TBBS Menu Commands

UltraChat offers the ability to invoke TBBS menu commands from within the chat environment. These commands, called DOTBBS commands, may be familiar to existing TBBS sysops, since the capability exists in other TBBS option modules. These commands allow you to tie an UltraChat command keystroke to the execution of a menu command type, by TBBS menu type number, just as you would from a TBBS menu. Up to nine (9) DOTBBS commands can be defined.

DOTBBS#xx_CMD	Command
DOTBBS#xx_TYPE	Other
DOTBBS#xx_DATA	Other

Where "xx" is the action command number from 01 to 09 (see below for details).

DOTBBS#01_CMD – DOTBBS#09_CMD

Execute a TBBS command. The DOTBBS#01_CMD through DOTBBS#09_CMD commands are all functionally identical. They allow you to designate up to nine (9) command to invoke TBBS menu commands. The menu command type (TBBS Type = num-

ber) and Opt Data to use for the command are specified using the appropriate command elements, discussed below.

IMPORTANT NOTES REGARDING THE USE OF THESE COMMANDS:

- A DOTBBS#xx_CMD command that normally would not return to UltraChat (any other option module, or types 5, 10, 12, 35, 43, or 45) are going to be marked as the user still being in UltraChat in any who's online displays.
- Eventually, a return to UltraChat using the /E:4 Opt Data switch entry style should be performed. This will place the user back where they were when they executing the command, and will displayed all the messages queued up while they were out of chat.
- DOTBBS#xx_CMD commands will exit a user from one-to-one private chat before executing.
- DO NOT EXECUTE A DOTBBS#xx_CMD COMMAND FROM WITHIN THE TBBS READ PROMPT! THIS WILL CAUSE SYSTEM MALFUNCTION, INCLUDING (BUT NOT LIMITED TO) SYSTEM CRASHES!

IMPORTANT: Anything that the user would normally have seen displayed while inside UltraChat is going to be queued while they are in a DOTBBS command. For that reason, whatever happens should be kept short and sweet.

DOTBBS#xx TYPE

This element specifies the TYPE = number of the TBBS command to execute.

DOTBBS#xx DATA

This element specifies the Opt Data for the TBBS command to execute.

Moan Macros

Moan macros in UltraChat are similar to “action words” in Major BBS chat. A moan macro makes a certain word yield canned messages while in public chat.

You define moan macros using a special command, `MOAN_MACRO`, and a variety of command elements that are sent dependent on several criteria.

A moan macro command is structured as follows:

```
MOAN_MACRO <word>
    DEFAULT_RESPONSE      <string>
    [DEFAULT_SENDS        <string>]
    [IF_USER_RESPONSE     <string>]
    [IF_USER_SENDS_USER   <string>]
    [IF_USER_SENDS        <string>]
    [IF_MSG_RESPONSE      <string>]
    [IF_MSG_SENDS         <string>]
    [IF_USER&MSG_RESPONSE <string>]
    [IF_USER&MSG_SENDS_USER <string>]
    [IF_USER&MSG_SENDS    <string>]
```

The `<word>` is the moan macro text itself, and can be any single word of 10 characters or less. The `DEFAULT_RESPONSE` element is required; all others are optional.

To understand what the various elements do requires some explanation of how moan macros function. First, some definitions:

default moan - user types just the moan macro word and nothing else (i.e., “yell”).

user moan - user types the moan macro followed by a user name and nothing else (i.e., “yell sysop”).

msg moan - user types the moan macro followed by text that is not a user name (i.e., “yell this stinks!”).

user&msg moan - user types the moan macro followed by a user name and additional text (i.e., “yell sysop this stinks!”).

DEFAULT RESPONSE <string>

Default response (response). The <string> is sent to the person that used the moan macro as a default moan. This string is also sent to the user of the moan macro if the appropriate *_RESPONSE string is not present for the type of moan macro used. This string is REQUIRED for all moan macros.

DEFAULT SENDS <string>

Default remote display (remote display). The <string> is sent to all other users in the channel if the moan macro was of the default moan type. This string is optional, and if not present when a default moan is used, the text typed will be transmitted to the conference as though it was not a moan macro at all.

IF USER RESPONSE <string>

Response when user name or ID given (response). The <string> is sent to the person that used the moan macro as a user moan. This string is optional, and if not present when a user moan is used, the DEFAULT_RESPONSE will be substituted.

IF USER SENDS USER <string>

Remote display for specific user when user name or ID given (remote display). The <string> is sent to the user specified in a user moan. This string is optional and if not present, the user will instead receive the IF_USER_SENDS string.

IF USER SENDS <string>

Remote display for all others when user name or ID given (remote display). The <string> is sent to all other users in the conference except the user that is the object of the user moan. This string is optional, and if not present when a user moan is used, UltraChat will instead convert the moan into a msg moan.

IF MSG RESPONSE <string>

Response when message given (response). The <string> is sent to the person that used the moan macro as a msg moan. This string

is optional, and if not present when a `msg` moan is used, the `DEFAULT_RESPONSE` will be substituted.

IF MSG SENDS <string>

Remote display when message given (remote display). The `<string>` is sent to all other users in the conference when a `msg` moan is used. This string is optional, and if not present when a `msg` moan is used, the text typed will be transmitted to the conference as though it was not a moan macro at all.

IF USER&MSG RESPONSE <string>

Response when user name or ID and message given (response). The `<string>` is sent to the person that used the moan macro as a `user&msg` moan. This string is optional, and if not present when a `user&msg` moan is used, the `DEFAULT_RESPONSE` will be substituted.

IF USER&MSG SENDS USER <string>

Remote display for specific user when user name or ID and message given (remote display). The `<string>` is sent to the user specified in a `user&msg` moan. This string is optional and if not present, the user will instead receive the `IF_USER&MSG_SENDS` string.

IF USER&MSG SENDS <string>

Remote display for all others when user name or ID given (remote display). The `<string>` is sent to all other users in the conference except the user that is the object of the `user&msg` moan. This string is optional, and if not present when a `user&msg` moan is used, UltraChat will instead convert the moan into a `msg` moan.

The following table will help show what strings get sent for various types of moans (strings are listed in precedence from top to bottom or left to right).

***** INSERT TABLE HERE *****

Using Text In UltraChat Command Elements

Command elements in UltraChat allow you to specify text. This text is shown to the user executing a command, shown to a user at the receiving end of a command, shown to the user to prompt them for input, or shown to the user in an error condition. In all cases, you specify text with the various command elements – this is their purpose. The general syntax is:

`<element_name> <string>`

Where `<element_name>` is the name of the command element (such as `CHAT#01_EXIT`) and `<string>` is the text to associate with that element.

Text for command elements can be formatted a variety of different ways. The text always follows these rules:

- Text must be contained in quotation marks.
- When double quotes are used to contain text, then embedded single quotes are allowed.
- When single quotes are used to contain text, then embedded double quotes are allowed.
- Text can flow from line-to-line if you follow the methods outlined later.
- All carriage returns must be provided explicitly – UltraChat does not insert any by itself. UltraChat will word wrap text displays automatically, however.
- Text strings, as placed in the configuration file, must be less than 500 characters in length. When expanded by UltraChat (insertion parameters are inserted where needed), the text cannot be longer than 1,000 characters.

Wrapping Text Line-to-Line

Text in command elements can span multiple lines if you desire. In order for UltraChat to handle this, however, you must format the text properly.

The first line of the command element is handled just as though you were having text on only one line, except that there is no terminating quotation mark. Subsequent lines must begin with a plus sign (+) as the first non-space character, which alerts UltraChat that the line is a continuation of a previous line.

You can use a split bar (|, also sometimes called a "pipe") to allow leading spaces in the follow-on text. If you omit the split bar, then UltraChat will display the text beginning with the first non-blank character that follows the plus sign.

Here is an example of a command element that spans multiple lines:

```
CHAT#01_REMOTE '^G&RUSER:C% is requesting you join
them in chat!^M
+   Type "C" from the menu or "chat %RUSER:C%"
+ | if already in chat to accept.^M'
```

Adding Carriage Returns and Control Characters

As mentioned previously, any text in a command element must explicitly contain a carriage return (also called a "new line") if you want one to be sent to the user's screen. If you fail to put a carriage return at the end of command elements, the next thing UltraChat displays will be displayed immediately following (butted up against) the last text displayed. Sometimes this is what you want to occur – other times it is not. In any case, where you want UltraChat to begin or end a command element's text display with a carriage return, you must put one in.

Carriage returns are specified with the characters ^M. That's a caret (^) followed by the letter M. This means Ctrl-M, which is a carriage return in the ASCII character set.

Here's an example of the use of the carriage return:

```
HELP#01_ERROR "Help file not found!^M"
```

The command element (an error message in this case) text begins with a double quote, followed by the text of the message, followed by ^M, followed by a double quote. This command element will be sent with a trailing carriage return when sent to a user. Carriage returns can come at the beginning or be embedded in the middle of text, too.

Other control characters can be specified the same way. You could embed a "bell" (Ctrl-G) character in the above text like so:

```
HELP#01_ERROR "^G^GHelp file not found!^M"
```

This example begins with a double quote, followed by a pair of ^G sequences before the rest of the text. This will send two bells, and cause most users' terminals to beep twice when this text is sent.

Here is a table of control characters:

^A = SOH	^I = Tab	^Q = DC1	^Y = EM
^B = STX	^J = LnFeed	^R = DC2	^Z = SUB
^C = ETX	^K = VTab	^S = DC3	^[= Escape
^D = EOT	^L = FmFeed	^T = DC4	^\\ = FS
^E = ENQ	^M = Enter	^U = NAK	^] = GS
^F = ACK	^N = SO	^V = SYN	^^ = RS
^G = Bell	^O = SI	^W = ETB	^_ = US
^H = Bksp	^P = DLE	^X = CAN	^@ = Null

Adding ANSI Escape Sequences

UltraChat allows you to colorize command element text displays with ANSI escape sequences, just like TBBS itself. ANSI escape sequences always begin with an Esc (Escape) character, followed by a left square bracket, followed by the contents of the ANSI sequence. Toward the end of Chapter 2 of the TBBS manual, you will find a complete ANSI escape sequence chart you can use to look-up possible ANSI codes.

In UltraChat (as well as most areas of TBBS itself), you designate an Escape character with a carat and a left square bracket, like this:

^[

ANSI escape sequences then have the following appearance within the UltraChat configuration file:

```
^[[1;36m
```

The carat and first left square bracket mean "Escape." The second left square bracket is a part of every ANSI escape sequence. The remainder of the sequence is the ANSI command contents, in this case a color (attribute) change command.

Restricting Configurations Within UltraChat

Like many eSoft products (TIMS, QSO, InterChange, etc.), UltraChat provides restrict blocks that allow you to control to whom any portion of the configuration applies. For example, you can have some UltraChat commands available only to some users (sysops, co-sysops, super-users, etc.). Or, provide a named conference for users who call on certain lines only. You have complete control over configurations via restrict blocks.

If you have used restrict blocks in other eSoft products, you already know how to use them in UltraChat. A restrict block begins with a **RESTRICT** verb, and ends with an **ENDRESTRICT** verb. The general form of the construct is as follows:

```
RESTRICT <restriction> ...  
(normal UltraChat config stmnts here)  
ENDRESTRICT
```

RESTRICT statements can be nested to any number of levels. This allows changing UltraChat configuration based on any or all of several restriction parameters.

IMPORTANT NOTE: IF A STRING (ie, something like WHO#01_TEXT) IS CHANGED, ONLY THE FIRST ONE THAT IS FOUND VALID FOR A USER WILL BE USED! This is important because in some cases a user may qualify under multiple restrict statements, but only the first occurrence of a string will be stored and used for that user. Because of this, put your **MOST** restrictive statements at the top of the file, and the least restrictive at the end of the file. For example, **SYSOP** restrictions might go first, while **ALL** strings might have a default that goes last

in the file and will be used if no other text was previously defined for that type.

The various `<restriction>` types are:

**A1 = <flag> A2 = <flag> A3 = <flag>
A4 = <flag>**

These four restrict types allows you to restrict based on TBBS access flags, where `<flag>` is of the usual SDL format (ie, A1=X...X.X).

P# = <param>

This restrict type allows you to restrict based on a parameter passed to UltraChat from the Opt Data line of the menu entry that invoked UltraChat. The # is 1-9 to match the parameter number passed to UltraChat. The `<param>` parameter contains text that much match the passed parameter exactly. Use quotation marks around the `<param>` text if it contains embedded spaces. Parameters can be up to 16 characters in length.

GROUP = <group>

This restrict type allows you to restrict based on group. The `<group>` is a 4 character identifier that should match the /G:abcd that this person was last given (ie, GROUP=MEN). Groups are defined with the Opt Data line of the menu entry that invoked UltraChat.

NAME = <name>

This restrict type allows you to restrict based on user name or ID. The `<name>` is a user name and this RESTRICT statement is only valid for that particular user. In most cases this will be used as NAME=SYSOP to give special functions to the sysop, although it can easily be used to give a certain user a special thrill.

LINE = <lines>

This restrict type allows you to restrict based on TBBS line number. The <lines> can be a single line number, a range of line numbers (i.e., LINE = 1-8) or any combination of line numbers and ranges separated by commas (i.e., LINE = 1,3-6,9,11-22).

LANG = <c>

This restrict type allows you to restrict based on TBBS language selection. The <c> is a character corresponding to the language you wish to match. If the language the user has enabled is NOT <c>, then the RESTRICT is not used. Languages are lettered A, B, C, etc.

PRIV <conditional> <priv_level>

This restrict type allows you to restrict based on a user's privilege level. The <conditional> is any combination of <, > and = to form things like > = for greater than or equal to, < > for not equal to, and so on. The <priv_level> is the privilege value that should be used for comparison.

For example, PRIV > = 50 will pass for anyone with a privilege greater than or equal to 50. By nesting RESTRICT blocks, ranges can be created, for example:

```
RESTRICT PRIV>=50
    RESTRICT PRIV<=100
        (stmts here for 50<=priv<=100)
    ENDRESTRICT
ENDRESTRICT
```

BAUD <conditional> <baud_rate>

This restrict type allows you to restrict based on the baud (bps) rate a user is connected at. The <conditional> is as it is for PRIV (discussed previously), and <baud_rate> is any valid TBBS baud (bps) rate that can be set in CEDIT. The result codes from the right side of the CEDIT result code table is used if it exists, otherwise the left side is used. Valid baud rates are currently things like 2400, 9600, and 19200 – but NOT things like 12000, 14400, or 28800. One

special case is that the local console is assumed to be baud rate 999999.

DAYS = <day>

This restrict type allows you to restrict based on the day of the week. The <day> parameter can be SUN, MON, TUE, WED, THU, FRI, or SAT, corresponding to the days of the week from Sunday through Saturday. You can also use the day names in verbose form, such as SUNDAY or THURSDAY for readability.

Two special "days" also exist: WEEKEND and WEEK, which correspond to SAT and SUN for WEEKEND, and MON through FRI for WEEK.

DATE = <date>

This restrict type allows you to restrict based on calendar date. The <date> parameter is of the format MM/DD/YYYY, where MM is the numeric month from 1 to 12, DD is the numeric day from 1 to 31, and YYYY is the year in standard 4-digit format (e.g., 1994).

You can replace any of the digits normally expected with an asterisk (*) which, like a DOS wildcard, means "any." For example:

```
RESTRICT DATE=04/01/****
```

Will pass on April Fool's Day (April 1st) in any year.

```
RESTRICT DATE=**/01/****
```

Will pass on the first of any month, any year.

Nesting Restrict Blocks

RESTRICT blocks can be nested, allowing you to create restrictions like the following:

```

RESTRICT A1=XXX-----
    <statements for all A1=XXX----- people>
    RESTRICT A2=...-----
        <stmnts for A1=XXX----- A2=...,----->
    ENDRESTRICT
    RESTRICT A2=XXX-----
        <stmnts for A1=XXX----- A2=XXX----->
    ENDRESTRICT
ENDRESTRICT

```

Because RESTRICT blocks can become very long if a lot of types of restriction are used, it is often necessary to split them onto multiple lines of text. To do that, simply start the next line with RESTRICT and make sure the proper number of ENDRESTRICT statements are also in place. This works because:

```

RESTRICT <restrict1> <restrict2>
ENDRESTRICT

```

is the same as:

```

RESTRICT <restrict1>
    RESTRICT <restrict2>
    ENDRESTRICT
ENDRESTRICT

```

Nested RESTRICTS are the same as using an "AND" statement, while sequential RESTRICT statements are separate entities.

Insertion Parameters

Insertion parameters play an integral role in virtually all command elements in the UltraChat configuration. Insertion parameters allow text displays to show user names, chat messages, and much more. What good would a who's online display be if it didn't show user names? Insertion parameters allow you to configure who's online displays so they have user names. What good would a facility to send a private chat message to somebody be if the user receiving the command couldn't see anything? Here again, insertion parameters allow text displayed by UltraChat to show this information and much more.

One of the most important concepts of insertion parameters to bear in mind is that of perspective. In TBBS, you never have to worry about this – in UltraChat, it is of critical importance.

Within the UltraChat configuration, all command element text is from the perspective of the user printing the text string. For example, let's say user A sends a private chat message to user B. The configuration is from the perspective of user A for text strings printed to him, and from the perspective of user B for text strings printed to him.

User B in this scenario is called the target user or remote user, and receives a remote display.

Insertion parameters must be carefully used then to ensure that the right user receive the right type of information. For example, in this scenario, the %LINE% insertion parameter would return user A's line number, while %RLINE% would return user B's line number when the string is printed to user A, the reverse is true for strings printed to user B.

Note the letter "R" in the insertion parameter's name – this means remote and can be used as a rule of thumb to denote insertion parameters that refer to the remote or target user in any given command scenario.

Insertion Parameter Formatting

Insertion parameters are of the form:

`%PARAM[:[+/-]D[U,L,C]]%`

This means that the parameter can be optionally followed by a format specifier. If no format specifier is present, the parameter is printed unchanged and takes a length equal to whatever length the parameter ends up being. If a format specifier is present, it is signaled by the parameter being followed by a colon (:), then an optional justification parameter.

The default justification parameter is LEFT justified and corresponds to a minus (-) symbol in this position. If a plus (+) is found in this position, the string is filled on the left to make it RIGHT justified.

Next is the length of the string. The default if not present is to end the string without padding it. If a number is present here, the insertion parameter will be truncated or padded to fit the given length. By default, if the string needs to be padded, it will be padded with blank spaces.

If the string length parameter starts with a 0, the string will be padded with 0's instead. The last optional parameter is one of U, L, or C, which corresponds to the "case" of the final result. By default, no alphabetic case translation is done to the string. If U is specified, the string is converted to all upper case, L converts to all lower case, while C converts to a mixed case format.

General Insertion Parameters

<code>%USER%</code>	The user's full name.
<code>%TIMELEFT%</code>	The amount of time left for this call.
<code>%NOTES%</code>	The user's NOTES field from the userlog.
<code>%ADDR1%</code>	The user's ADDR1 field from the userlog.
<code>%ADDR2%</code>	The user's ADDR2 field from the userlog.
<code>%CITY%</code>	The user's CITY field from the userlog.

Chapter 3: Command Reference

%STATE%	The user's STATE field from the userlog.
%ZIP%	The user's ZIP field from the userlog.
%PHONE%	The user's PHONE field from the userlog.
%COUNTRY%	The user's COUNTRY field from the userlog.
%LOCATION%	The user's location.
%3LINEBIO%	The user's three-line bio.
%LASTON%	The user's last logon date and time as: MM/DD/YY HH:MM.
%LOGINS%	The number of times the user has logged on.
%TIMEUSED%	The amount of time a user has been logged on NOT counting the current call. It is in hours:MM format where hours can be up to 5 digits long.
%FLAGS%	Prints strings for WHO#02_TEXT, WHO#03_TEXT and WHO#04_TEXT if user qualifies.
%ANNOUNCE%	The user's announce string (if any).
%LINE%	The user's line number.
%CHANNEL%	The user's current channel number. If the user is in a named conference, the conference name is displayed.
%MINCHANNEL%	Lowest public channel number allowed
%MAXCHANNEL%	Highest public channel number allowed.
%SYSTEM%	The SYSTEM_NAME_TEXT string (BBS name).
%SYSTEMID%	The SYSTEM_ID_TEXT string (BBS ID text).
%SYSOP%	The SYSOP_NAME_TEXT string (sysop's name).
%LISTED%	Number of lines listed in WHO#01_CMD or WHO#02_CMD display.
%LASTLINE%	Number of lines in system.

%MSG%	Chat message to be sent (used by private paging and chat commands, among others).
%TYPED%	What the user typed in its entirety, designed to allow an entire user's typed command to be echoed back if desired.
%IS/ARE%	Valid in "totals" lines for WHO commands and prints IS_TEXT or ARE_TEXT depending on if the number of users listed by the command was 1 or more than 1.
%VERSION%	Returns name of program and current version number in the form: <div style="text-align: center;"><code>ultrachat Version 1.00<.xx></code></div> where the .xx will show mod level when mods are issued.
%TOPIC%	User's private channel topic string (if any).
%PASSWORD%	User's private channel password.
%P1% - %P9%	User's passed parameters 1 through 9.
%NUMCMDS%	Number of UltraChat commands available for "this" user (independent of how many may be configured).
%FREETEXT%	The amount of free memory space available to hold UltraChat commands for "this" user. Useful for diagnostic purposes.
%TIME%	The current clock time in HH:MM format.
%DATE%	The current calendar date in MM/DD/YY format.
%EURODATE%	The current calendar date in DD/MM/YY format.
%WAITING%	Displays the contents of the WAITING_TEXT configuration statement only if a user has mail in TBBS waiting for them.
%IGNORED%	Displays the contents of the IGNORED_TEXT configuration statement only if the user being referred to is being ignored.
%INVITED%	Displays the contents of the INVITED_TEXT configuration statement online if the user being referred to has been invited to a private channel.

Remote (Target) User Insertion Parameters

The following insertion parameters refer to the "current" remote line. In the case of who's online displays, this is the information for the next line to be displayed by the who's online display command. In other cases, such as user to user private messages, it refers to the remote user or the user who is the target of the command, as outlined previously.

%RUSER%	Remote user's full name.
%RNOTES%	Remote user's NOTES field from the userlog
%RADDR1%	Remote user's ADDR1 field from the userlog
%RADDR2%	Remote user's ADDR2 field from the userlog
%RCITY%	Remote user's CITY field from the userlog
%RSTATE%	Remote user's STATE field from the userlog
%RZIP%	Remote user's ZIP field from the userlog
%RPHONE%	Remote user's PHONE field from the userlog
%RCOUNTRY%	Remote user's COUNTRY field from the userlog
%RLOCATION%	Remote user's location
%R3LINEBIO%	Remote user's three-line bio
%RLASTON%	Remote user's last logon date and time as: MM/DD/YY HH:MM.
%RLOGINS%	Remote number of times the user has logged on.
%RTIMEUSED%	Remote amount of time a user has been logged on NOT counting the current call. It is in hours:MM format where hours can be up to 5 digits long.
%RFLAGS%	Prints strings for WHO#02_TEXT, WHO#03_TEXT and WHO#04_TEXT if remote user qualifies.
%RANNOUNCE%	Remote user's announce string (if any).

%R HANDLE %

%RCHANNEL%	Remote remote user's channel if in chat (# or P# if in private chat). If the user is in a named conference, the conference name is displayed.
%RLINE%	Remote user's line number.
%RCOMMAND%	%RCOMMAND% [rcommand]Remote user's command being run. Will print "CONF: <name>" for named conference.
%RMSG%	Message sent from the other line.
%RTOPIC%	Remote user's private channel topic string (if any).
%RP1% - %RP9%	Remote user's passed parameters 1 through 9.

Userlog Image Insertion Parameters

The following insertion parameters work on a userlog image from disk. The userlog record used for these is that of the user initiating the related command. Any command that prints things from the userlog (WHO#03_CMD for example), should use these parameters for printing the desired information:

%UUSER%	The user's full name.
%UNOTES%	The user's NOTES field from the userlog.
%UADDR1%	The user's ADDR1 field from the userlog.
%UADDR2%	The user's ADDR2 field from the userlog.
%UCITY%	The user's CITY field from the userlog.
%USTATE%	The user's STATE field from the userlog.
%UZIP%	The user's ZIP field from the userlog.
%UPHONE%	The user's PHONE field from the userlog.
%UCOUNTRY%	The user's COUNTRY field from the userlog.
%ULOCATION%	The user's location.

Chapter 3: Command Reference

%U3LINEBIO%	The user's three-line bio.
%ULASTON%	The user's last logon date and time as: MM/DD/YY HH:MM.
%ULOGINS%	The number of times the user has logged on.
%UTIMEUSED%	The amount of time a user has been logged on NOT counting the current call. It is in hours:MM format where hours can be up to 5 digits long.
%UFLAGS%	<p>Prints strings for WHO#02_TEXT, WHO#03_TEXT and WHO#04_TEXT if user qualifies.</p> <p>See the sample UCHAT.CTL file for how these parameters can be used.</p>

Glossary

- action commands** This term refers to a set of special commands within the chat environment. Action commands are a form of generic moan macro. In Major BBS chat, this is usually called a "generic action" or "GA" command. In essence, an action command allows a user to create on-the-fly a special, tailored remote display that will be shown to all the users who are in the channel with the user who issued the action command. In most ways an action command is the same as a user simply typing something into the chat channel - but the remote display is different, formatted as though the user issued a moan macro.
- announce string** The announce string is a user-settable vanity item that can be shown as part of who's online displays (or other UltraChat displays). It is often applied as a customized greeting or remark by a user.
- blo** See three-line bio.
- chat environment** This term refers to any point at which the user is under the control of UltraChat. Who's online displays, public chat, private chat, and so on all collectively constitute the chat environment. Any function a user performs online with TBBS that is not under the control of UltraChat is not a part of the chat environment.
- chat traffic** This term refers to the messages, pages, private comments, and other commands issued by a user which result in a display being made on another user's screen. While within the chat environment, anything that a user types that results in a display on the screen(s) of one or more users is considered chat traffic. Not all operations within the chat environment create chat traffic. For example, a who's online display is not considered chat traffic because the results of the command are shown only to the user issuing the command, and not to anyone else.
- command** A series of one or more keystrokes issued by a user within the chat environment to cause UltraChat to do something. Commands include such things as who's online displays, channel change, paging, private messages, and so on.
- command element** The text associated with a particular UltraChat command. Elements include responses, remote displays, prompts and error messages.
-

Appendix A: Glossary

command element sharing	A technique for economizing the use of memory by having two or more command elements share the identical text.
error message	This is one of the primary command elements in UltraChat. An error message is shown to a user issuing a chat command when the command is improperly formatted, references a user not logged in, etc., to alert the user to the fact that they made a mistake in the issuance of the command.
groups	<p>This term refers to an UltraChat feature that allows you to separate users into logical groups. When you force a user to become part of a group, that user can only interact with other users in his group for any UltraChat related functions: who's online displays will only show users in his group; he will be able to page only users in his group; in a public channel a user can only chat with people in his own group; and so on.</p> <p>IMPORTANT: Groups do not apply to users who are not online. Who's online and other displays which reference information from the userlog file (any insertion parameter that begins with a %U) will display the user whether they would normally be a part of the group or not. To limit display of this information, you must use the MATCH_USERLOG_FLAG configuration statement to control this outside of the grouping construct.</p>
handle	This term refers to an identifier that overrides the name or ID of a user within the chat environment. In essence, a handle is a replacement user name or ID set exclusively for use within chat. It is primarily of interest to systems who require users to logon with their own real names, but want to offer anonymity to their users when they use chat. A handle will be used for any displays and commands within the chat environment, such as who's online displays, paging commands, three line bio displays, etc.
ignore	This term applies to the state other users can be in within the chat environment from the perspective of a particular user. When a user is inside the chat environment, they can mark specific other users as being ignored. When another user is ignored, their chat traffic is not displayed until they are remembered again. All the ignored users' chat traffic is suppressed, including private messages and public messages as applicable, as if the ignored user wasn't sending any chat traffic. Other users in the chat environment still see the traffic of the ignored user (unless they too explicitly ignore that user). This is the opposite of remember.

input buffer	This term refers to a memory buffer (a storage area in memory) where keystrokes from a user are stored before being acted upon by TBBS. Whenever a user presses a key during a BBS session, it is placed in the input buffer. Whenever TBBS is able to process a key, it removes the first one available in the input buffer. Ordinarily, this processing of the input buffer takes mere microseconds, and is totally transparent to the user.
invisible	This term applies to a particular state a user can be in within the chat environment. When a user is marked as invisible, they will not show-up on who's online displays (unless configured specifically to show-up there), and will otherwise be treated by UltraChat as though they are not online. The state of the invisible setting can be forced upon a user (as it often is for new users to your BBS), or if you allow it, can be user settable (a user can mark themselves invisible or visible).
invite	This term applies to private channels, and refers to a situation where a user permits / requests that another user join him in his own private channel. When another user is invited, they are permitted to change to the inviting user's private channel at their leisure. This is the opposite of uninvite.
no chat	This term, also given in some cases as the single word "nochat", refers to a particular state that a user can be in within the chat environment. When a user is marked as being in no chat mode, UltraChat will internally understand that the user is not available for chat. As such, private messages, paging, and other chat traffic is not sent to a user who is in no chat mode.
one-shot	This term refers to the execution of a specific, single UltraChat function from a TBBS menu entry. When the one-shot is executed, the UltraChat function runs, then returns control to TBBS immediately afterward. This allows UltraChat operations, such as a handle change, user page, who command, etc., to be executed from a menu entry without entering a public chat section.
private message queue	This term refers to a storage place for private messages. UltraChat creates a private message queue on a user-by-user basis. When private messages are received for a user (private messages being things like chat pages, logon announcements, etc.), but the user is not within a window of opportunity, the messages are saved in the private message queue. The next time the user passes a window of

opportunity, the private messages that have been queued up are then displayed to the user's screen.

prompt This is one of the primary command elements in UltraChat. It refers to a prompt or question given to a user in conjunction with that user's issuance of a chat command. Prompts simply request additional information from the user issuing the command, such as which user the command affects, what message to send, and so on. For example, a user issuing a page command might see a prompt to enter the name of the user they wish to page.

remember The opposite of ignore; see "ignore" for details.

remote display This is one of the primary command elements in UltraChat. It refers to text shown to a user (or users) who is the "target" of a chat command issued by another user. For example, if user A send a page command to user B, user B will receive a remote display that indicates that user A is paging him. See also: response.

response This is one of the primary command elements in UltraChat. It refers to text shown to a user as a result of that user's issuance of a chat command. Responses generally serve to confirm with the user that UltraChat received and processed the command. See also: remote display.

target This term applies to many UltraChat commands, and refers to the user who is at the receiving end of a command. For example, if user A pages user B, user B is said to be the target of the command. Not all commands in UltraChat have a target, for example, a command to exit the chat facilities has no target. In rare cases, the target may be multiple users (for example, a user issues an invite command to "ALL" instead of a specific user).

three-line bio This term applies to a feature of UltraChat whereby users can place information about themselves. In most situations, the three-line bio contains personal statistics about the user; such as height, weight, etc.; but since the user enters the three-line bio themselves it can contain anything the user wishes. The three-line bio is stored by UltraChat, and can be made available to other users with specific commands, on who's online displays, etc.

typeahead This term refers to keystrokes that a user has made, but which TBBS (and/or UltraChat) have not yet acted upon. These are keys that

	have stacked up, i.e., keys that have been typed ahead, hence the term "typeahead."
uninvite	This term applies to private channels, and refers to a situation where a user no longer permits or kicks out another user from his own private channel. When another user is uninvited, they are no longer permitted to change to the uninviting user's private channel, and if they are already in that private channel, they are removed (kicked out). This is the opposite of invite.
visible	The opposite of invisible; see "invisible" for details.
window of opportunity	<p>This term applies to moments in the passage of time during a user's BBS session where they are brought into the chat environment for a brief moment in order to receive chat traffic. When within that window, a user can receive chat traffic such as logon announcements, private paging messages, etc. When outside the window, a user cannot receive chat traffic. For example, a user who is running a TDBS application or downloading a file is not within the window of opportunity, and therefore, cannot receive any chat traffic from UltraChat until they again enter a window of opportunity. Windows of opportunity include a brief moment when TBBS loads a menu, and anytime a user is sitting at the message read prompt while reading messages.</p> <p>NOTE: Whenever a user is within chat, whether public or private, they are also inside a sort of large, extended window of opportunity and can receive all chat traffic at any time while in chat. When outside of chat, chat traffic is only received when the user passes through a window of opportunity. In chat, the window is only open when at the prompt, and not while the user is typing a command.</p>

Technical Limits of UltraChat

The following technical limits apply to UltraChat:

Each defined command can have up to 14 characters of command text (the command the user types to execute the command).

Named conferences can have names up to 30 characters long.

A DOTBBS command can use a maximum of 100 characters of Opt Data.

There can be a maximum of 12,000 bytes of text strings active for a user. This is combined sum of the length of all command strings and command response strings (but does not include moan macros).

A maximum of 65,536 numbered conferences can be defined.

A maximum of 65,536 named conferences can be defined, of which, only 50 can be active for a user at any one given time.

Moan macros can be a maximum of 10 characters in length.

A maximum of 512 moan macros can be made available.

Parameter strings (those designated with the /Pn: <string> Opt Data switch) can be a maximum of 16 characters in length.

A maximum of 10 lines (users) can be "highlighted" with the highlight command at any one time.

Announce strings can be a maximum of 80 characters in length.

A three-line bio can be a maximum of 255 characters in length.

A user's private channel topic (the name he gives for his private channel) can be a maximum of 24 characters in length.

Passwords for private channels can be a maximum of 16 characters in length.

Handles can be a maximum of 20 characters in length.

Appendix B: Technical Limits

The user command portion of a who's online display (customized text to define the place a user is online, set with the /D: Opt Data switch) can be a maximum of 30 characters in length.

Index

A

ABORT script command	4-10
aborting (killing) input	
message displayed when	3-25
ACCEPT script command	4-10
access flags	
restricting by	3-68
accessing UltraChat	1-8
action commands	3-58
action words	
See moan macros	
ACTION#01_CMD	3-59
ACTION#02_CMD	3-59
ACTION#03_CMD	3-59
ACTION#04_CMD	3-59
ACTION#05_CMD	3-59
ACTION#06_CMD	3-59
ACTION#07_CMD	3-59
ACTION#08_CMD	3-59
ACTION#09_CMD	3-59
ACTION#xx_CONFIRM	3-59
ACTION#xx_REMOTE	3-59
%ADDR1%	3-73
%ADDR2%	3-73
%ANNOUNCE%	3-6, 3-74
announce strings	3-6
displaying	3-74
setting or changing	3-51
ANNOUNCE_CHARS_NUM	3-6
ANSI escape sequences	3-26
adding to text	3-66
answering no	
key used to	3-8
answering yes	
key used to	3-8
ARE_TEXT	3-15

B

baud rate	
restricting by	3-69
BAUD script command	4-11
BBS name	
setting	3-14
BIO#01_CMD	3-50
BIO#02_CMD	3-50
BIO#xx_PROMPT	3-50
BIO#xx_YN	3-50
BLANK	2-18
usage of	3-3
browse a user	3-37

C

/C: switch	1-12
CALC script command	4-11
carat	3-65
carriage returns	
adding to text	3-65
changing channels	3-27
changing handle	3-52
%CHANNEL%	3-74
channel numbers	
displaying	3-74
invalid entered display	3-29
prompting for	3-29
setting highest public	3-7
setting maximum	3-7
setting minimum	3-6
CHANNEL#01_CMD	3-27, 3-29 - 3-30
CHANNEL#02_CMD	3-27
CHANNEL#03_CMD	3-28, 3-30
CHANNEL#04_CMD	3-28 - 3-30
CHANNEL#05_CMD	3-28
CHANNEL#06_CMD	3-28
CHANNEL#07_CMD	3-28
CHANNEL#xx_BADNUM	3-29
CHANNEL#xx_BADPW	3-30
CHANNEL#xx_CHANPMT	3-29

CHANNEL#xx_CONFIRM.....	3-29
CHANNEL#xx_ENTRY.....	3-29
CHANNEL#xx_EXIT.....	3-29
CHANNEL#xx_NOTINV.....	3-30
CHANNEL#xx_NOTNUM.....	3-30
CHANNEL#xx_PWPPMT.....	3-29
channels	
changing.....	3-27
changing to private.....	3-28
confirmation of change.....	3-29
message displayed on entry to.....	3-29
returning back to original.....	3-28
returning to.....	1-9
setting highest public.....	3-7
setting maximum.....	3-7
setting minimum.....	3-6
chat	
private one-to-one.....	3-19
public.....	3-19
chat exit commands.....	3-41
chat message	
displaying.....	3-75
CHAT#01_CMD.....	3-31 - 3-33
CHAT#01_TEXT.....	3-17
CHAT#02_CMD.....	3-31 - 3-32
CHAT#02_TEXT.....	3-17
CHAT#03_CMD.....	3-31
CHAT#03_TEXT.....	3-17
CHAT#04_CMD.....	3-31
CHAT#04_TEXT.....	3-17
CHAT#05_CMD.....	3-32
CHAT#05_TEXT.....	3-18
CHAT#06_CMD.....	3-32
CHAT#06_TEXT.....	3-18
CHAT#07_TEXT.....	3-18
CHAT#08_TEXT.....	3-18
CHAT#09_TEXT.....	3-19
CHAT#10_TEXT.....	3-19
CHAT#11_TEXT.....	3-19
CHAT#12_TEXT.....	3-19
CHAT#13_TEXT.....	3-19
CHAT#14_TEXT.....	3-20
CHAT#15_TEXT.....	3-20
CHAT#16_TEXT.....	3-20

Index

CHAT#17_TEXT	3-20
CHAT#18_TEXT	3-20
CHAT#19_TEXT	3-20
CHAT#20_TEXT	3-21
CHAT#21_TEXT	3-21
CHAT#22_TEXT	3-21
CHAT#23_TEXT	3-21
CHAT#24_TEXT	3-21
CHAT#xx_CONFIRM	3-32
CHAT#xx_EXIT	3-32
CHAT#xx_NOCHAT	3-33
CHAT#xx_PROMPT	3-32
CHAT#xx_REMOTE	3-33
%CITY%	3-73
code stacking	
of option modules	1-5
color	
adding to text	3-66
color changing configuration statements	3-26
COLOR#01_TEXT	3-26
COLOR#02_TEXT	3-26
COLOR#03_TEXT	3-26
COLOR#04_TEXT	3-26, 3-55
command elements	
adding color to	3-66
connecting to commands	2-10
definition of	3-1
description of	2-9
error messages	2-9
formatting of	3-3, 3-64
prompts	2-9
remote displays	2-9
responses	2-9
sharing of	3-4
using text in	3-64
wrapping text in	3-65
commands	
action	3-58
chat exit	3-41
connecting to command elements	2-10
definition of	3-1
description of	2-9
edit	3-51
formatting of	3-2

general chat.....	3-30
help file display.....	3-42
interaction.....	3-53
keystrokes for.....	2-9
naming convention.....	2-10
private message.....	3-33
shortcut.....	3-9
status forced off.....	3-48
status forced on.....	3-46
status toggle.....	3-44
text display.....	3-43
three-line bio.....	3-49
who's online.....	3-36
conferences	
named.....	1-10
configuration commands	
channel changing.....	3-27
general chat text.....	3-17
configuration files	
compilation of.....	1-3
creating.....	2-12
including others.....	3-6
introduction to customizing.....	2-8
samples.....	1-15
configuration statements	
color changing.....	3-26
general.....	3-5
general error text.....	3-22
miscellaneous text.....	3-23
text.....	3-14
user related.....	3-16
configurations	
controlling.....	3-12
reference for.....	3-1
restricting.....	3-67
sample.....	2-8
connecting commands and elements.....	2-10
control characters	
adding to text.....	3-65
converting from other BBS software.....	1-16
%COUNTRY%.....	3-74
creating configuration file.....	2-12
customization	
introduction to.....	2-8

D

/D: switch.....	1-13
date.....	3-75
restricting by	3-70
day of the week	
restricting by	3-70
DEFAULT_RESPONSE	3-62
DEFAULT_SENDS	3-62
DEFINE_CONF	2-14, 3-7
DELAY script command	4-11
DIAL script command	4-12
directory	
temporary	1-2
displays	
pausing	3-8
stopping	3-8
DLX	2-1, 2-8
emulation	1-15
style of entry	1-9
DOTBBS#01_CMD	3-60
DOTBBS#02_CMD	3-60
DOTBBS#03_CMD	3-60
DOTBBS#04_CMD	3-60
DOTBBS#05_CMD	3-60
DOTBBS#06_CMD	3-60
DOTBBS#07_CMD	3-60
DOTBBS#08_CMD	3-60
DOTBBS#09_CMD	3-60
DOTBBS#xx_DATA	3-60
DOTBBS#xx_TYPE	3-60

E

/E:1 switch.....	1-9
/E:2 switch.....	1-9
/E:3 switch.....	1-9
/E:4 switch.....	1-9
/E:5 switch.....	1-9
/E:6 switch.....	1-10, 2-14
ECHO script command	4-12
edit commands	3-51

EDIT#01_CMD.....	3-6, 3-51
EDIT#02_CMD.....	3-51
EDIT#03_CMD.....	3-52
EDIT#04_CMD.....	3-52
EDIT#xx_CONFIRM.....	3-52
EDIT#xx_ERROR.....	3-52
EDIT#xx_PROMPT.....	3-52
EMS memory.....	1-5
emulations	
sample.....	1-15
END script command.....	4-12
ENDRESTRICT.....	3-67
ENTER_STRING.....	3-11
entry	
of three-line bio.....	1-8
to a specific channel.....	1-9
to last channel.....	1-9
to named conference.....	1-10
to public chat.....	1-9
to UltraChat.....	1-9
error messages.....	2-9
ERROR#01_TEXT.....	3-22
ERROR#02_TEXT.....	3-22
ERROR#03_TEXT.....	3-22
ERROR#04_TEXT.....	3-22
ERROR#05_TEXT.....	3-22
ERROR#06_TEXT.....	3-23
ERROR#07_TEXT.....	3-23
ERROR#08_TEXT.....	3-23
eSoft	
technical support.....	1-18
%EURODATE%.....	3-75
EXACT.....	2-16
usage of.....	3-3
executing TBBS menu commands	
from chat.....	3-59
EXIT#01_CMD.....	2-15, 3-41
EXIT#02_CMD.....	2-16, 3-42
EXIT#03_CMD.....	3-42
EXIT#04_CMD.....	3-42
exiting chat	
message displayed when.....	3-24

F

%FLAGS%	3-74
features of UltraChat	2-3
FidoNet	1-16
%FLAGS%	3-16
formatting	
of command elements	3-3
of commands	3-2
%FREETEXT%	3-75
functionality	
palette of	2-11
functions	
accessing as one-shot	1-12
accessing from menus	1-12

G

/G: switch	1-10
general chat text configuration commands	3-17
general configuration statements	3-5
general error text configuration statements	3-22
generic action commands	3-58
glossary	A-1
GOTO script command	4-12
GRAB script command	4-13
groups	1-10
ID for	1-11
maximum number of	1-11
restricting by	3-68

H

handles	
changing	3-52
forcing the use of	1-10
HANGUP script command	4-13
help file display commands	3-42
HELP#01_CMD	3-43
HELP#02_CMD	3-43
HELP#03_CMD	3-43

HELP#04_CMD	3-43
HELP#05_CMD	3-43
HELP#06_CMD	3-43
HELP#07_CMD	3-43
HELP#08_CMD	3-43
HELP#09_CMD	3-43
HELP#xx_ERROR	3-43
HELP#xx_FILE	3-43
highlighting chat traffic	3-55
HOT_KEYS	3-11

I

/I switch	1-7
/I:1 switch	1-8
/I:2 switch	1-8
/I:3 switch	1-8
IF script command	4-14
IF_MSG_RESPONSE	3-62
IF_MSG_SENDS	3-63
IF_USER&MSG_RESPONSE	3-63
IF_USER&MSG_SENDS	3-63
IF_USER&MSG_SENDS_USER	3-63
IF_USER_RESPONSE	3-62
IF_USER_SENDS	3-62
IF_USER_SENDS_USER	3-62
%IGNORED%	3-15, 3-75
IGNORED_TEXT	3-15
ignoring users	3-53 - 3-54
inactivity timeout	3-10
expiration of	3-25
INACTIVITY_TIMER	3-10
INCLUDE	3-6
including other config files	3-6
initialization	
of UltraChat	1-7
input	
aborting (killing) of	3-10
insertion parameters	3-72
formatting of	3-73
general	3-73
perspective of	3-72
remote (target user)	3-76

userlog image	3-77
installation of UltraChat	1-2
INTERACT#01_CMD	3-53
INTERACT#02_CMD	3-53
INTERACT#03_CMD	3-54
INTERACT#04_CMD	3-54
INTERACT#05_CMD	3-54
INTERACT#06_CMD	3-55
INTERACT#07_CMD	3-55
INTERACT#xx_ALLOFF	3-56
INTERACT#xx_ALLON	3-56
INTERACT#xx_ERROR	3-58
INTERACT#xx_NAMEPMT	3-57
INTERACT#xx_OFF	3-56
INTERACT#xx_OFF2	3-56
INTERACT#xx_ON	3-55
INTERACT#xx_ON2	3-55
INTERACT#xx_REMOTE	3-57
INTERACT#xx_REMOVE	3-57
interaction commands	3-53
introduction to UltraChat	1-1
invisible users	
In who's online displays	3-13
%INVITED%	3-15, 3-75
INVITED_TEXT	3-15
inviting users to private chat channel	3-54
%IS/ARE%	3-14 - 3-15, 3-75
IS_TEXT	3-14
K	
key	
used to answer no	3-8
used to answer yes	3-8
used to enter public chat	3-8
used to exit public chat	3-8
used to pause display	3-8
used to stop display	3-8
keystroke	
suppressing on exit	1-13
KILL_LINE_KEY	2-13, 3-10 - 3-11

L

/L[G]: switch.....	4-4
language selection	
restricting by.....	3-69
LAST_LINE_NUM.....	3-7
%LASTLINE%.....	3-74
%LASTON%.....	3-74
learning UltraChat.....	2-8, 2-12
limits	
technical.....	B-1
%LINE%.....	3-74
line numbers	
controlling whether used in commands.....	3-9
forcing use in commands.....	3-9
restricting by.....	3-69
setting last used on displays.....	3-7
link commands.....	4-2
LINK#01_CMD.....	4-2
LINK#01_CONFIRM.....	4-2
LINK#01_NAMEPMT.....	4-2
LINK#02_CMD.....	4-3
LINK#03_CMD.....	4-4
linking	
configuration.....	4-2
establishing.....	4-2
establishing a link.....	4-4
Introduction to.....	4-1
manual outdial.....	4-6
script commands.....	4-10
script language for.....	4-7
scripted outdial.....	4-7
LIST_INVISIBLE_SWITCH.....	3-13
LIST_ME_SWITCH.....	2-13, 3-13
LIST_PRIVATE_SWITCH.....	3-13
%LISTED%.....	3-74
%LOCATION%.....	3-74
%LOGINS%.....	3-74
login announcements	
acknowledgement display.....	3-25
initialization.....	1-8
suppressing.....	1-8

text used for	3-24
LOOP script command	4-16
M	
macros	
moan	3-61
Major BBS	2-8, 3-6, 3-11, 3-58
and linking	4-1
emulation	1-15
MATCH_ONLINE_FLAG	3-12
MATCH_USERLOG_FLAG	3-12
MAX_CHANNEL_NUM	2-13, 3-7
%MAXCHANNEL%	3-74
memory	
amount taken	1-4
EMS	1-5
not enough to install	1-2
OM CODE	1-4
OM UDATA	1-4
option module code stacking	1-5
usage of	1-4
when there's not enough	1-5
menus	
creating for UltraChat	1-6
top level	1-7
MESSAGE script command	4-16
MIN_CHANNEL_NUM	2-13, 3-6
%MINCHANNEL%	3-74
MISC#01_TEXT	3-23
MISC#02_TEXT	3-24
MISC#03_TEXT	3-24
MISC#04_TEXT	3-24
MISC#05_TEXT	3-24
MISC#06_TEXT	3-24
MISC#07_TEXT	3-24
MISC#08_TEXT	3-10, 3-25
MISC#09_TEXT	3-25
MISC#10_TEXT	3-25
MISC#11_TEXT	3-10, 3-25
MISC#12_TEXT	3-10, 3-25
MISC#13_TEXT	3-10, 3-25
MISC#14_TEXT	3-10, 3-25

MISC#15_TEXT	3-10, 3-25
MISC#16_TEXT	3-10, 3-26
miscellaneous text configuration statements	3-23
moan macros	3-61
MODE =	2-15
usage of	3-2
%MSG%	2-20, 3-75
multilingual mode	
effects on UltraChat	1-7

N

/N switch	1-13
name	
restricting by	3-68
NAME_NUM_PARSER	3-9
named conferences	1-10, 2-14
defining	3-7
entry to	1-10
naming convention	
for commands	2-10
nesting RESTRICT blocks	3-71
new users	
numbered considered new	3-6
NEW_USERS_NUM	3-6
no chat mode	3-16, 3-35
NO_KEY	3-8
%NOTES%	3-73
%NUMCMDS%	3-75

O

OFF#01_CMD	3-48
OFF#02_CMD	3-48
OFF#03_CMD	3-48
OFF#04_CMD	3-9, 3-49
OFF#05_CMD	3-49
ON#01_CMD	3-46
ON#02_CMD	3-47
ON#03_CMD	3-47
ON#04_CMD	3-9, 3-47
ON#05_CMD	3-47
ON#xx_TEXT	3-47

one-shots	1-12
suppressing key on exit from	1-13
optimizing performance	2-21
option module code stacking	1-5
Oracomm	1-12, 2-1, 2-8, 3-6, 3-8, 3-17 - 3-19, 3-31
emulation	1-15
style of entry	1-9
other BBS software	
converting from	1-16
OVERALL_TIMER	3-10

P

%P1%	3-75
%P2%	3-75
%P3%	3-75
%P4%	3-75
%P5%	3-75
%P6%	3-75
%P7%	3-75
%P8%	3-75
%P9%	3-75
/P: switch	1-11
palette of functionality	2-11
parameters	
insertion	3-72
passing to UltraChat	1-11
restricting by	3-68
parsing (processing)	
mixed style	3-9
of chat commands	3-9
TBBS style	3-9
PARTIAL	2-15
usage of	3-3
%PASSWORD%	3-75
passwords	
for private chat channels	3-52
prompting for	3-29
PATTERN script command	4-16
PAUSE script command	4-17
PAUSE_KEY	3-8
pausing display	
key used for	3-8

performance	
optimizing	2-21
perspective	
of insertion parameters	3-72
%PHONE%	3-74
pipe	3-65
plus sign	3-65
press any key prompt	3-24
private channels	
passwords for	3-28
toggling in and out	3-27
private chat	
exiting from	3-42
how displayed in who's online commands	3-13
private chat channel	
setting password for	3-52
setting topic (name) for	3-51
private message commands	3-33
private message queue	
display of	3-11
setting display mode	3-11
where stored	3-5
private messages	
confirmation of delivery	3-34
resending	3-34
sending	3-34
sending to a linked system	4-4
private one-to-one chat	3-19
PRIVATE#01_CMD	2-19, 3-34
PRIVATE#02_CMD	3-34
PRIVATE#03_CMD	3-34
PRIVATE#04_CMD	3-34
PRIVATE#05_CMD	3-34
PRIVATE#06_CMD	3-34
PRIVATE#07_CMD	3-34
PRIVATE#08_CMD	3-34
PRIVATE#09_CMD	3-34
PRIVATE#10_CMD	3-34
PRIVATE#xx_CMD	3-22
PRIVATE#xx_CONFIRM	3-34
PRIVATE#xx_EMPTY	3-35
PRIVATE#xx_ERROR	3-36
PRIVATE#xx_MSGPMT	3-35
PRIVATE#xx_NAMEPMT	3-35

PRIVATE#xx_NOCHAT	3-35
PRIVATE#xx_REMOTE	3-35
privilege level	
restricting by	3-69
prompts	2-9
PUB_CHANNEL_NUM	3-7
public chat	3-19
entering	3-8, 3-31
exiting from	3-8
public messages	
acknowledging	3-20
format of	3-17
PUBLIC_ENTRY_KEY	3-8
PUBLIC_EXIT_KEY	3-8

Q

QSO	1-16
queue	
See private message queue	
QWK offline readers	1-16

R

%RADDR1%	3-76
%RADDR2%	3-76
RAM disk	1-3
%RANNOUNCE%	3-6, 3-16, 3-76
RAWXMIT script command	4-17
%RCHANNEL%	3-77
%RCITY%	3-76
%RCOMMAND%	1-13, 3-77
%RCOUNTRY%	3-76
read-only mode	3-13
READONLY_SWITCH	3-13
relationships	
between commands and elements	2-10
RELEASE script command	4-17
remembering users	3-53
remote displays	2-9
resending private messages	3-34
responses	2-9

RESTRICT blocks.....	3-67
nesting of.....	3-71
restricting	
by access flags.....	3-68
by baud rate	3-69
by date.....	3-70
by day of the week	3-70
by groups	3-68
by language selection.....	3-69
by line	3-69
by parameters	3-68
by privilege level.....	3-69
by user name.....	3-68
configurations.....	3-67
%RFLAGS%.....	3-16, 3-76
RIP kit	1-7
%RLASTON%	3-76
%RLINE%	2-20, 3-77
%RLOCATION%.....	3-76
%RLOGINS%	3-76
%RMSG%	3-77
%RNOTES%	3-76
%RP1%	3-77
%RP2%	3-77
%RP3%	3-77
%RP4%	3-77
%RP5%	3-77
%RP6%	3-77
%RP7%	3-77
%RP8%	3-77
%RP9%	3-77
%RPHONE%	3-76
%RSTATE%	3-76
%R3LINEBIO%.....	3-76
%RTIMEUSED%.....	3-76
%RTOPIC%	3-77
%RUSER%	2-20, 3-76
%RZIP%.....	3-76

S

sample emulations.....	1-15
------------------------	------

script language	
for linking.....	4-7
scripting	
of linking.....	4-7
scripts	
debugging of.....	4-21
sending private messages.....	3-34
session script command.....	4-18
SET script command.....	4-18
sharing of command elements.....	3-4
shortcut commands.....	3-9
SHORTCUT_KEY.....	3-9
SPEED script command.....	4-19
split bar.....	3-65
%STATE%.....	3-74
statements	
definition of.....	3-1
status forced off commands.....	3-48
status forced on commands.....	3-46
status toggle commands.....	3-44
STOP_KEY.....	3-8
stopping display	
key used for.....	3-8
suppressing keystroke on exit.....	1-13
%SYSOP%.....	3-14, 3-74
sysop name	
displaying.....	3-74
setting.....	3-14
SYSOP_NAME_TEXT.....	2-13, 3-14
%SYSTEM%.....	3-14, 3-74
system name	
displaying.....	3-74
SYSTEM_ID_TEXT.....	2-13, 3-14
SYSTEM_NAME_TEXT.....	2-13, 3-14
%SYSTEMID%.....	3-14, 3-74

T

TBBS

benefits.....	1-16
command line.....	1-2
emulation.....	1-15
menus for UltraChat.....	1-6

message read prompt.....	3-24, 3-31
style of command parsing	3-9
top (main) menu.....	1-7
versions supported	1-2
TBBS menu commands	
executing from chat	3-59
TBBS_PARSER	2-13, 3-9
TBSPATH = setting	2-22
TDBS.....	1-16
technical limits.....	B-1
temporary directory	1-2, 3-5
text	
adding carriage returns to	3-65
adding color	3-66
adding control characters to	3-65
using in command elements	3-64
wrapping line to line	3-65
text configuration statements.....	3-14
text display commands.....	3-43
%3LINEBIO%	3-74
three-line bio	
and initialization.....	1-8
commands.....	3-49
displaying	3-74
suppressing.....	1-8
time	3-75
limiting in chat	3-10
time limit	
expiration of.....	3-25
message displayed when expired	3-25
%TIMELEFT%.....	3-15, 3-73
timer	
inactivity.....	3-10
message displayed when expired	3-25
overall	3-10
TIMER script command	4-19
%TIMEUSED%	3-74
TIMS.....	1-16
TOGGLE#01_CMD	3-44
TOGGLE#02_CMD	3-45
TOGGLE#03_CMD	3-45
TOGGLE#04_CMD	3-9, 3-45
TOGGLE#05_CMD	3-45
TOGGLE#xx_OFF	3-46

TOGGLE#xx_ON.....	3-46
%TOPIC%.....	3-75
tutorial.....	2-12
creating configuration file.....	2-12
entering chat.....	2-14
exiting chat.....	2-14
listing who's online.....	2-17
private chat.....	2-19
summary of.....	2-21
TYPE#01_CMD.....	3-44
TYPE#02_CMD.....	3-44
TYPE#03_CMD.....	3-44
TYPE#04_CMD.....	3-44
TYPE#05_CMD.....	3-44
TYPE#06_CMD.....	3-44
TYPE#07_CMD.....	3-44
TYPE#08_CMD.....	3-44
TYPE#09_CMD.....	3-44
TYPE#10_CMD.....	3-44
TYPE#11_CMD.....	3-44
TYPE#12_CMD.....	3-44
TYPE#13_CMD.....	3-44
TYPE#14_CMD.....	3-44
TYPE#15_CMD.....	3-44
TYPE#16_CMD.....	3-44
TYPE#17_CMD.....	3-44
TYPE#18_CMD.....	3-44
TYPE#19_CMD.....	3-44
TYPE#xx_TEXT.....	3-44
%TYPED%.....	3-75

U

%UADDR1%.....	3-77
%UADDR2%.....	3-77
UCHAT.*** file.....	2-22
UCHAT.CTL.....	
See configuration files.....	
UCHAT_DIR.....	2-13, 3-5
UCHATMON.*** file.....	2-22
%UCITY%.....	3-77
%UCOUNTRY%.....	3-77
%UFLAGS%.....	3-78

%ULASTON%	3-78
%ULOCATION%	3-77
%ULOGINS%	3-78
UltraChat	
accessing	1-8
configurability of	2-3
configuration reference	3-1
customization of	2-8
entry to (from menu)	1-9
general features	2-3
initialization of	1-7
learning	2-8, 2-12
optimizing performance	2-21
orientation	2-3
temporary directory	3-5
tutorial	2-12
why eSoft created	2-1
uninviting users	
from private chat channel	3-54
message displayed when	3-24
UNLIMITED_TEXT	3-15
%UNOTES%	3-77
UPDATE program	1-2
%UPHONE%	3-77
%USER%	3-73
user browsing	3-37
user related configuration statements	3-16
USER#01_TEXT	3-16
USER#02_TEXT	3-16
USER#03_TEXT	3-16
USER#04_TEXT	3-16
userlog image insertion parameters	3-77
users	
customizing where they are display	1-13
forcing read-only	3-13
grouping	1-10
highlighting chat traffic	3-55
ignoring	3-53 - 3-54
inviting to private chat channel	3-54
listing in who's online displays	3-13
matching command sets	3-12
remembering	3-53
separating into groups	1-10
uninviting from private chat channel	3-54

waiting to chat.....	3-17
users waiting to chat.....	3-18
%USTATE%.....	3-77
%U3LINEBIO%	3-78
%UTIMEUSED%	3-78
%UUSER%	3-77
%UZIP%	3-77

V

%VERSION%	3-75
-----------------	------

W

WAIT script command	4-19
WAITFOR script command.....	4-19
%WAITING%	3-14, 3-75
WAITING_TEXT	3-14
WHEN script command.....	4-20
where a user is	
customizing.....	1-13
WHO#01_CMD.....	2-18, 3-37
WHO#02_CMD.....	2-18, 3-37
WHO#03_CMD.....	3-37
WHO#04_CMD.....	3-37
WHO#05_CMD.....	3-37
WHO#06_CMD.....	3-6, 3-37
WHO#07_CMD.....	2-19, 3-37, 3-39
WHO#08_CMD.....	3-37
WHO#09_CMD.....	3-37
WHO#10_CMD.....	3-37
WHO#11_CMD.....	3-38
WHO#12_CMD.....	3-38
WHO#13_CMD.....	3-38
WHO#xx_ANON.....	3-41
WHO#xx_CMD.....	3-7, 3-12
WHO#xx_ERROR.....	3-41
WHO#xx_HEADER	3-38
WHO#xx_HISCHAN.....	3-40
WHO#xx_IDLE	3-39
WHO#xx_INVISIBLE	3-39
WHO#xx_MYCHAN	3-40
WHO#xx_NAMEPMT	3-41

WHO#xx_NORMAL.....	3-38
WHO#xx_NOUSERS.....	3-40
WHO#xx_PAUSE.....	3-39
WHO#xx_PRIVATE.....	3-40
WHO#xx_PUBLIC.....	3-39
WHO#xx_TOTAL.....	3-38
who's online commands.....	3-36
who's online displays.....	3-13
custom user-dependent displays.....	3-16
customizing.....	1-13
from a linked system.....	4-3
setting highest line number.....	3-7
wrapping text in command elements.....	3-65
 X	
XMIT script command.....	4-21
 Y	
YES_KEY.....	3-8
 Z	
%ZIP%.....	3-74
